

Multi-level Temporal Understanding in Video Analysis: From Action Recognition to Quality Assessment

by

Yaoxin Li

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
System Design Engineering

Waterloo, Ontario, Canada, 2025

© Yaoxin Li 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Svetlana Yanushkevich
Professor, Schulich School of Engineering
Department of Electrical and Software Engineering
University of Calgary

Supervisor(s): Alexander Wong
Professor, Systems Design Engineering
University of Waterloo

Internal Member: John Zelek
Associate Professor, Systems Design Engineering
University of Waterloo

Internal Member: Bryan Tripp
Associate Professor, Systems Design Engineering
University of Waterloo

Internal-External Member: William Melek
Professor, Mechanical and Mechatronics Engineering
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Chapter 3 incorporates research from the publication "Spot the Difference! Temporal Coarse to Fine to Finer Difference Spotting for Action Recognition in Videos" [54], which was accepted at the IEEE International Conference on Multimedia and Expo (ICME) and designated as a Best Paper Candidate. As the primary author of this work, I was responsible for the conceptualization and design of the model architecture, experimental methodology, implementation and training of the computational models, and preparation and submission of the manuscript. My supervisor, Professor Alexander Wong, provided strategic guidance throughout the research process and contributed to the editorial refinement of the manuscript prior to publication.

Abstract

Artificial intelligence algorithms have permeated virtually every facet of contemporary life, from personalized shopping recommendations and targeted advertising to search engine optimization and multimedia content delivery. Among these applications, video content—as a predominant carrier of information in multimedia—occupies a position of exceptional significance. However, the application of AI algorithms to video analysis remains insufficiently mature, with comprehensive video content interpretation persisting as a critical challenge. This dissertation presents an integrated framework for multi-level temporal understanding in video analysis, advancing a coherent progression from fundamental action recognition to temporal localization and ultimately to qualitative assessment.

The foundation of our integrated approach begins with addressing fundamental limitations in current action recognition paradigms. We identify a critical gap in existing methodologies where trimmed approaches restrict model training to curated action segments, while untrimmed methods rely solely on video-level supervision. Both approaches fail to exploit the inherent complementarity between action and non-action segments within complete temporal sequences. To address this limitation, we introduce a novel multi-stage contrastive learning architecture that hierarchically extracts motion-critical features through coarse-to-fine-to-finer temporal contrasting. This approach establishes a self-correcting learning regime where action discriminability emerges from explicit comparisons against non-action references, effectively suppressing static bias amplification while enhancing temporal sensitivity.

Building directly upon this temporal understanding foundation, our second research direction extends these temporal insights to the challenge of precise action localization within continuous video streams. This integrated framework incorporates a multi-modal fusion classifier with adaptive modality weighting, a Class-Semantic Attention mechanism for precise proposal generation, and cross-domain prototype alignment enabling knowledge transfer between trimmed and untrimmed paradigms. This advancement represents a natural progression from recognizing what actions occur to precisely determining when they occur within the temporal dimension. This system was validated through participation in the CVPR ActivityNet Temporal Action Localization challenge, achieving second-place rankings in both 2021 and 2022 editions, with the action classification component attaining first place on the validation set—demonstrating the practical efficacy of our theoretical contributions.

The third component completes our multi-level temporal understanding framework by addressing how well actions are performed—moving beyond recognition and localization to

qualitative assessment. This progression reflects the natural evolution of video understanding from basic detection to increasingly nuanced temporal analysis. We develop a novel Mamba-based framework that leverages selective state space models to capture long-range dependencies in sequential data. This approach enables fine-grained analysis of temporal patterns and subtle motion nuances that determine action quality across various domains including sports performance, physical rehabilitation, and skill assessment. Experimental results demonstrate superior performance compared to transformer and CNN-based methods in both accuracy and computational efficiency.

The collective contributions of this dissertation establish a comprehensive framework for multi-level temporal understanding in videos. Rather than representing isolated research directions, these three components form a coherent progression from fundamental action recognition (what), to precise temporal localization (when), and finally to qualitative assessment (how well). This integrated approach advances both the theoretical understanding and practical capabilities in AI-driven video analysis, offering a unified perspective that bridges the gap between academic research and real-world applications across the spectrum of video understanding tasks. By addressing these complementary facets within a unified framework, our work establishes new foundations for temporal video understanding that can be applied across diverse application domains.

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Alexander Wong, for his invaluable guidance and unwavering support, especially during the more challenging moments of my PhD journey. His insights and encouragement have been instrumental in shaping my research. I am also incredibly grateful to my boyfriend, Xuanqi Chen, for always being by my side, offering patience, understanding, and unwavering support through every high and low.

Additionally, I want to thank everyone in the VIP lab for fostering a stimulating and supportive research environment. The insightful discussions and collaborations have made this journey both enriching and enjoyable. I am also grateful to my colleagues from my internship, whose guidance and shared expertise have provided me with invaluable learning experiences and professional growth opportunities. Working alongside them has expanded my perspective and deepened my understanding of real-world applications of AI.

Lastly, I am forever grateful to my parents—without their support and sacrifices, from coal mines to steel plants, I wouldn't have the opportunity to pursue AI today. Their dedication and resilience continue to inspire me every day.

Table of Contents

Examining Committee	ii
Author’s Declaration	iii
Statement of Contributions	iv
Abstract	v
Acknowledgements	vii
List of Figures	xiv
List of Tables	xvi
1 Introduction	1
1.1 Motivation	2
1.1.1 The Evolution of Digital Communication Eras	2
1.1.2 The Scale and Computational Challenge of Video Content	4
1.1.3 The Multi-Level Nature of Human Video Understanding	4
1.2 Core Challenges in Video Understanding	5
1.2.1 Challenge 1: Motion Temporal Variability in Action Recognition . .	5
1.2.2 Challenge 2: Context-Dependency in Temporal Action Localization	6

1.2.3	Challenge 3: Multi-Modal Integration in Action Quality Assessment	8
1.3	Research Scope and Contributions	9
1.3.1	Research Scope Definition	9
1.3.2	Primary Research Contributions	10
1.3.3	Broader Impact and Applications	11
1.4	Organization of this Thesis	12
2	Background	14
2.1	Introduction	14
2.2	Overview of Video Understanding Tasks	15
2.3	Evolution of Deep Learning Architectures in Video Understanding	20
2.3.1	The Computational Challenge of Video Understanding	20
2.3.2	From 2D to 3D Convolutional Paradigms	22
2.3.3	Temporal Modeling with Recurrent and Attention Mechanisms	30
2.3.4	Transformer Revolution in Video Understanding	33
2.3.5	State Space Models and the Mamba Paradigm	45
2.3.6	Future Directions and Hybrid Architectures	48
2.4	Action Recognition: Foundations and Evolution	49
2.4.1	Traditional Methods: The Era of Hand-crafted Features	50
2.4.2	Deep Neural Architectures in Video Understanding	51
2.5	Temporal Action Localization	59
2.5.1	Task Definition	59
2.5.2	Technical Evolution	60
2.5.3	Boundary Matching Networks: A Two-Stage Approach	62
2.6	Action Quality Assessment: Foundations and Evolution	64
2.6.1	Introduction and Problem Definition	65
2.6.2	Evolution of Technical Approaches	66
2.6.3	Challenges and Future Directions	67
2.7	Chapter Summary and Conclusion	68

3	Spot the Difference! Temporal Coarse to Fine to Finer Difference Spotting for Action Recognition in Videos	70
3.1	Introduction	70
3.1.1	The Core Challenge: Beyond Simple Classification	71
3.1.2	A New Perspective: Learning Through Discrimination	71
3.1.3	Key Terminology and Framework Components	72
3.1.4	Our Approach and Contributions	73
3.1.5	Chapter Organization	74
3.2	Temporal Discrimination Learning for Action Recognition	74
3.2.1	Cognitive Inspiration: The "Spot the Difference" Paradigm	74
3.2.2	From Cognitive Insight to Technical Innovation	75
3.2.3	Technical Background and Problem Formulation	76
3.2.4	Problem Definition and Method Overview	78
3.2.5	Model-Agnostic Integration and Advantages	79
3.2.6	Mathematical Formulation	80
3.3	Coarse Level: Discrimination Across Videos	81
3.3.1	Motivation and Problem Definition	82
3.3.2	Booster Networks: Definition and Design	82
3.3.3	Integration with Backbone Architecture	83
3.3.4	Training Objective and Loss Function	84
3.3.5	Information Flow and Feature Propagation	85
3.4	Fine Level: Discrimination Within a Video	85
3.4.1	Background: Contrastive Learning	86
3.4.2	Proposed Fine-Level Contrastive Learning	88
3.5	Finer Level: Discrimination Inside One Segment	92
3.5.1	Motivation and Challenges	92
3.5.2	Background: Cross-Modal Alignment Approaches	93

3.5.3	Proposed Clip Matching Mechanism	94
3.5.4	Clip to Video-Clip Similarity Matching	95
3.5.5	Integration and Advanced Analysis	98
3.5.6	Integration with Overall Framework	100
3.5.7	Benefits of Finer-Level Discrimination	100
3.6	Experiments and Results	101
3.6.1	Datasets	101
3.6.2	Implementation Details	103
3.6.3	Results	104
3.6.4	Ablation Study	110
3.7	Chapter Summary and Conclusion	125
4	Temporal Action Localization	127
4.1	Introduction	127
4.1.1	Problem Definition and Scope	127
4.1.2	Technical Approach and System Architecture	128
4.1.3	Experimental Validation and Contributions	130
4.1.4	Chapter Organization	130
4.2	Video Feature Extraction	131
4.2.1	Foundation Models and Pre-training Strategy	131
4.2.2	Coarse-to-Fine-to-Finer Discriminative Learning	131
4.2.3	Multi-Architecture Feature Extraction Pipeline	132
4.2.4	Self-Supervised Feature Integration	133
4.2.5	Strategic Feature Combination Methodology	133
4.3	Multi-modality Classification System	134
4.3.1	Training Trimmed Classifiers	134
4.3.2	Second Stage Classifiers Based on Features	135
4.3.3	Proposal-Weighted Classifiers	136

4.3.4	Classification Ensemble Process	137
4.3.5	Classifier Refinement Through Multi-modal Analysis	138
4.4	Temporal Action Proposal Generation	141
4.4.1	Feature Encoding for Action Proposal	141
4.4.2	Boundary-Matching Network Architecture	141
4.4.3	Integration with Classification System	142
4.4.4	Class Semantics-Based Attention (CSA)	142
4.4.5	Technical Formulation of CSA	144
4.4.6	Experimental Evaluation	146
4.4.7	Ablation Studies	152
4.4.8	Visualization of attention maps	156
4.5	Chapter Summary and Conclusion	156
5	Multimodal Selective State Space Learning for Efficient Action Quality Assessment	160
5.1	Introduction	160
5.1.1	The Central Challenge: Temporal Modeling for Quality Assessment	160
5.1.2	Our Solution: Mamba for Efficient Temporal Modeling	161
5.1.3	Chapter Contributions	162
5.2	Background and Motivation	163
5.2.1	Evolution of Action Quality Assessment Methods	163
5.2.2	State Space Models: Theoretical Foundation	165
5.2.3	Multimodal Integration Challenges	166
5.2.4	Research Gap and Motivation	166
5.3	Methodology	168
5.3.1	Problem Formulation	168
5.3.2	Multimodal Selective State Space Learning	168
5.4	Experimental Results	173

5.4.1	Dataset and Evaluation Metrics	173
5.4.2	Implementation Details	173
5.4.3	Performance on Rhythmic Gymnastics Dataset	174
5.4.4	Ablation Studies	175
5.4.5	Computational Efficiency Analysis	176
5.5	Chapter Summary and Conclusion	177
6	Conclusion	179
6.1	Introduction	179
6.2	Summary of Contributions	180
6.3	Limitations and Lessons Learned	182
6.4	Future Work	183
6.4.1	Technical Advancements in Video Understanding	184
6.4.2	Scaling and Efficiency Considerations for Future Video Understanding	184
6.4.3	Integration with Large Language Models and Video-Language Models	185
6.5	Broader Impact and Final Thoughts	186
	References	188

List of Figures

1.1	Evolution of digital content across technological eras	3
1.2	Core components of video understanding technology	10
2.1	Overview of key video understanding tasks	16
2.2	Evolution of video understanding architectures (2014-present)	23
2.3	Comparison of 2D and 3D convolution operations	26
2.4	SlowFast Network architecture	29
2.5	Transformer architecture	34
2.6	Selective state space model architecture	46
2.7	Improved Dense Trajectories (iDT) feature extraction method	52
2.8	Temporal Shift Module architecture	55
2.9	Space-time attention mechanisms in TimeSformer	57
2.10	Classification of temporal action localization approaches	61
2.11	Boundary-Matching Network (BMN) architecture	63
3.1	“Spot the Difference” game example	75
3.2	Coarse-to-fine-to-finer action recognition framework	77
3.3	Multi-level discrimination framework for action recognition	79
3.4	Clip matching mechanism	95
3.5	Feature activation maps highlighting action-specific regions	123
3.6	Comparative feature activation analysis between our method and baseline	124

4.1	CSA attention mechanism for action detection	143
4.2	Qualitative Results on THUMOS-14	155
4.3	Comparison of Activation Maps	157
4.4	Channel Attention Profiles Across Videos	158

List of Tables

3.1	Performance comparison of SOTA architectures on ActivityNet	106
3.2	Performance comparison of SOTA architectures on HACS	107
3.3	Performance comparison of SOTA architectures on FineAction	108
3.4	Video-level performance comparison on ActivityNet	109
3.5	Ablation study of loss components on ActivityNet	111
3.6	Impact of positive and negative pair selection strategies on ActivityNet . .	114
3.7	Comparison of background awareness methods on ActivityNet	116
3.8	Impact of network architecture on clip matching performance	119
3.9	Impact of booster network architecture on ActivityNet	120
3.10	Action localization performance using features from our method	121
4.1	Multi-stage classification system performance on ActivityNet	140
4.2	Comparing our temporal localization method with SOTAs on THUMOS-14	148
4.3	Temporal action localization performance on ActivityNet v1.3	149
4.4	Impact of CSA module on proposal quality	150
4.5	Comparison of Attention Modules on THUMOS-14	152
4.6	CSA Module Kernel Size and Blocks on THUMOS-14	153
4.7	CSA Module Placement Ablation on THUMOS-14	154
4.8	Validation mAP on ActivityNet with BMN and BMN-CSA	155
5.1	Performance evaluation on Rhythmic Gymnastics dataset	174
5.2	Ablation Study on Rhythmic Gymnastics Dataset	175

Chapter 1

Introduction

The way humans communicate and share information has undergone a profound transformation with the emergence of digital technologies. From the earliest days of written language to the modern era of instantaneous global communication, each technological revolution has reshaped not only how we convey ideas but also how we perceive and process information [71]. This thesis examines the latest evolution in this journey—the rise of video as the dominant medium of digital expression [8]—and explores the computational challenges and opportunities it presents for artificial intelligence systems seeking to understand visual content at human-like levels [48].

Video understanding, defined as the automated computational process of interpreting and analyzing moving visual content to extract semantic meaning, identify temporal patterns, and understand contextual relationships, represents one of the most complex and multifaceted challenges in artificial intelligence research. To clarify this definition: **semantic meaning** refers to the conceptual content and significance of what is observed, **temporal patterns** are the changes and progressions that occur over time, and **contextual relationships** are the dependencies between different parts of the video that influence interpretation. Unlike static images or text, video incorporates temporal dynamics, multi-modal information streams, and contextual dependencies that more closely mirror how humans experience and interpret the world.

As video continues to proliferate across social media, education, entertainment, and professional environments, the development of sophisticated video understanding systems becomes increasingly crucial. These systems enable applications ranging from automated content moderation and recommendation systems to security surveillance, medical diagnosis, and autonomous vehicle navigation. This thesis presents a systematic investigation of

fundamental video understanding tasks through the lens of **multi-level temporal understanding**—a framework that progresses from basic action identification to sophisticated quality evaluation, mirroring the hierarchical nature of human video comprehension.

1.1 Motivation

1.1.1 The Evolution of Digital Communication Eras

The digital revolution has fundamentally reshaped human communication through three distinct evolutionary phases, each marked by transformative shifts in content creation and consumption patterns (shown in Figure 1.1). Understanding this evolution provides crucial context for why video understanding has become a central challenge in modern artificial intelligence and why multi-level temporal analysis is essential for processing contemporary digital content.

The early internet era (1990s–2000s) prioritized text-based interactions, constrained by limited **bandwidth** (the maximum rate of data transfer across a network connection, measured in bits per second) and computational power [13]. During this period, a typical dial-up internet connection provided only 56 kilobits per second, making video transmission impractical. Websites consisted primarily of text and simple graphics, with multimedia content being rare and requiring lengthy download times. This era established the foundation for digital information exchange but remained fundamentally limited in expressive capacity.

The advent of social media platforms in the 2000s–2010s ushered in an image-centric era, leveraging improved connectivity through broadband internet and mobile networks to enable visual storytelling [45]. Platforms like Instagram, Facebook, and Pinterest capitalized on increased bandwidth (typically 1-10 Mbps) and improved mobile cameras to make image sharing seamless and instantaneous. This era demonstrated the power of visual communication and established user behaviors around visual content consumption, setting the foundation for video’s eventual dominance.

Today, we inhabit a video-first paradigm, where high-speed mobile networks (particularly 4G LTE and emerging 5G networks providing data speeds exceeding 100 Mbps), ubiquitous smartphone access, and algorithmic content delivery have propelled video to the forefront of digital expression [8]. This transition reflects more than technological progress—it signifies a profound reconfiguration of human communication patterns, as dynamic visual narratives increasingly supplant static text and images in conveying complex

Digital Content Evolution: System Architecture

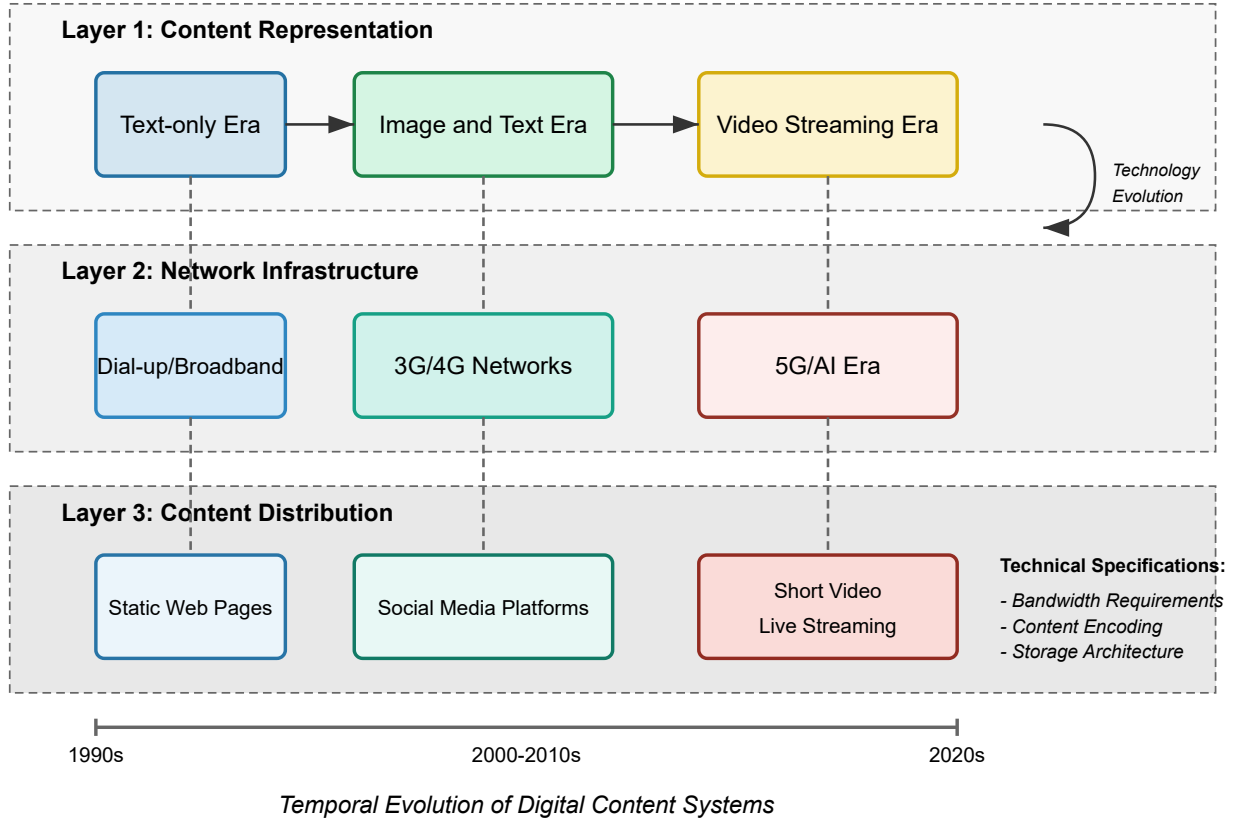


Figure 1.1: Technological epochs of digital content: Text-based communication (Dial-up/Broadband era), image-driven social media (3G/4G era), and video-dominated ecosystems (5G/AI era), illustrating the interplay between connectivity advances and content complexity.

ideas, emotions, and cultural values. The shift aligns with human cognitive preferences: research indicates the brain processes visual information 60,000 times faster than text, while multimodal stimuli combining motion, sound, and context enhance information retention by up to 95% compared to text-only content [72].

1.1.2 The Scale and Computational Challenge of Video Content

Video’s dominance as the primary medium of digital communication is both a cause and consequence of technological innovation. Several quantitative measures illustrate this dominance and its implications for computational analysis. Global mobile internet penetration has exceeded 67.5% as of 2024 [43], democratizing video production and consumption on an unprecedented scale. Video now constitutes over 82% of all internet traffic, with projections indicating this will surpass 95% by the end of 2025 as immersive formats like 360° video and volumetric capture gain traction [18].

The proliferation of affordable recording devices has enabled billions to create and share content instantaneously, generating a data volume that exceeds human analytical capacity. A single minute of HD video contains approximately 1,800 discrete frames, each with roughly 8 million pixels, creating unprecedented data density. When combined with audio waveforms, metadata, and potential multi-camera perspectives, the information complexity of video content renders traditional manual annotation and analysis methods obsolete at scale [88].

This seismic shift toward video content presents a critical paradox: while video offers unparalleled expressive capacity, its inherent complexity challenges both human and computational analysis. Unlike text’s linear structure or images’ static composition, video amalgamates spatial semantics, temporal dynamics, and multimodal context into a continuous data stream. The challenge extends beyond mere data volume to fundamental questions about how machines can achieve the holistic understanding that humans demonstrate effortlessly when interpreting dynamic visual content.

1.1.3 The Multi-Level Nature of Human Video Understanding

Human video comprehension operates through multiple levels of temporal understanding that seamlessly integrate to form coherent interpretation. When watching a soccer match, viewers simultaneously process basic visual elements (players, ball, field), recognize discrete actions (running, kicking, passing), understand temporal sequences (build-up play leading to scoring opportunities), and evaluate performance quality (technique, timing, tactical awareness). This multi-level processing happens naturally and instantaneously, representing a sophisticated integration of spatial analysis, temporal modeling, and contextual reasoning.

Consider the complexity involved when a human observer watches a tennis match. At the most basic level, they identify objects and their spatial relationships—players, rack-

ets, ball, court boundaries. At the action recognition level, they classify discrete activities—serving, volleying, running, celebrating. At the temporal localization level, they understand when significant events occur and how they relate to match progression—the start and end of each point, the sequence of shots within a rally. At the quality assessment level, they evaluate technical execution—the precision of a serve, the timing of a return, the strategic value of shot placement.

This natural human capability represents the gold standard for artificial video understanding systems. However, computational approaches have traditionally addressed these levels in isolation, missing the integrative processing that enables human-like comprehension. Current systems can often detect individual players or classify isolated actions, but struggle to correlate movement patterns with tactical significance, understand the temporal flow of game dynamics, or evaluate the quality of athletic performance in context.

The gap between human and machine capabilities in video understanding stems from the multi-faceted nature of video data. While significant progress has been made in narrow video tasks through **deep learning** approaches (machine learning methods using neural networks with multiple layers to learn hierarchical representations), the holistic understanding that humans achieve effortlessly remains elusive for machines. Current approaches often excel in isolated components while failing to grasp the narrative continuity that connects discrete events into meaningful sequences, highlighting the need for multi-level temporal understanding frameworks that can bridge these gaps.

1.2 Core Challenges in Video Understanding

This thesis addresses three fundamental challenges that represent critical barriers to achieving human-like video understanding capabilities. Each challenge corresponds to a specific level of temporal understanding and tackles distinct but foundational elements of the video understanding problem.

1.2.1 Challenge 1: Motion Temporal Variability in Action Recognition

In the domain of action recognition, despite considerable progress in recent years, significant challenges persist that limit the effectiveness of current approaches in real-world scenarios. **Motion temporal variability** refers to the phenomenon where identical actions exhibit dramatically different temporal patterns across different instances, contexts, or performers.

This challenge is particularly problematic in **action recognition**—the computational task of identifying and classifying activities occurring in video sequences.

Unlike image classification, which focuses on static visual features, action recognition requires modeling complex temporal dynamics across variable time scales. A professional tennis player’s serve involves explosive, precise movements completed in under a second, while an amateur’s serve might contain hesitations, repositioning, and slower execution of the same fundamental action. Similarly, running can exhibit varying stride lengths, arm movements, and speeds depending on terrain, fatigue level, or individual biomechanics. The human visual system effortlessly recognizes a “jumping” action whether performed by an Olympic athlete or a child, despite substantial differences in execution dynamics.

Current methodologies often develop **spurious correlations** with static contextual elements (background objects, scene setting) rather than capturing genuine motion patterns that define actions. This **static bias problem** leads to poor generalization when background contexts change, limiting real-world effectiveness. Models trained on tennis videos with specific court surfaces or lighting conditions may fail when applied to different environments, even though the fundamental action patterns remain consistent. Traditional **fixed-window approaches** (methods that analyze video using predetermined, unchanging time segments) struggle to differentiate between motion variations that change the action’s identity and stylistic differences in performing the same action.

Our research introduces a novel paradigm that explicitly contrasts action against non-action segments within video sequences. This approach fundamentally reframes the action recognition problem from one of simple classification to discriminative feature learning that highlights motion-critical information. By developing a hierarchical coarse-to-fine-to-finer framework, our work enables more robust action recognition that focuses on intrinsic motion characteristics rather than contextual cues. The methodology employs temporal contrast learning to distinguish between periods of meaningful action and background activity, forcing models to learn representations that capture essential motion dynamics rather than incidental scene elements.

1.2.2 Challenge 2: Context-Dependency in Temporal Action Localization

The second fundamental challenge addresses the precise temporal boundaries of actions within continuous video streams. **Context-dependency** represents the fundamental requirement that video segment interpretation depends critically on global temporal context.

Current approaches lack sufficient enhancement of global context features, leading to models that do not have a comprehensive understanding of the overall content and are unable to accurately detect action boundaries within video sequences.

The field of temporal action localization represents a critical extension of action recognition, focusing not only on what actions occur but precisely when they begin and end within **untrimmed videos** (raw footage containing both action segments and non-action background content). This task remains particularly challenging due to the inherent complexity of processing long, variable-length video sequences and the subtle transitions between action and non-action segments. Real-world videos rarely consist of continuous action; instead, they contain periods of relevant activity interspersed with transition phases, preparation periods, and background events that provide context but do not constitute the primary actions of interest.

The interpretation of any video segment fundamentally depends on understanding the global context of the entire video sequence. Without proper global context modeling, the same visual pattern may be misinterpreted as different actions, and models struggle to identify precise temporal boundaries where actions begin and end. A hand moving toward an object could indicate reaching, waving, or gesturing, but accurate interpretation requires understanding the global context of the video sequence. Similarly, a person’s body posture might indicate preparing to jump or recovering from a fall, but this determination depends on the global temporal context surrounding that moment.

Current attention mechanisms applied to temporal action localization tasks show limited effectiveness in this specialized application scenario, particularly in capturing the global context necessary for accurate boundary detection. Traditional approaches typically employ a **two-stage optimization** process that treats recognition and localization as separate problems, leading to suboptimal performance and computational inefficiency. The first stage identifies potential action segments, while the second stage classifies these segments independently. This separation fails to leverage the mutual dependencies between "what" and "when" questions—knowing precisely when an action occurs can improve classification accuracy, while better classification can lead to more precise temporal boundaries.

Our research addresses these limitations through a comprehensive system that integrates three key innovations: a multi-modal fusion classifier, a Class-Semantic Attention mechanism specifically designed for temporal action localization, and cross-domain prototype alignment. Our system employs a multi-modal classifier that fuses different visual features and uses other modalities as auxiliary information to enhance global context understanding and improve temporal boundary detection accuracy.

1.2.3 Challenge 3: Multi-Modal Integration in Action Quality Assessment

The third fundamental challenge moves beyond identifying and localizing actions to evaluating how well they are performed. **Multi-modal integration** addresses the challenge of combining different types of information streams (visual, audio, temporal, contextual) to achieve comprehensive understanding. This challenge is particularly critical in **action quality assessment**—evaluating how well actions are performed rather than simply identifying their occurrence.

In the emerging field of action quality assessment, we confront unique challenges including the subjective nature of quality judgments, the need for fine-grained temporal analysis, and the integration of domain-specific expertise into computational frameworks. Unlike action recognition, which typically involves discrete categorical decisions, quality assessment requires nuanced evaluation along continuous scales that reflect expert human judgment. The assessment criteria vary significantly across domains: a gymnastics routine evaluation focuses on form, difficulty, and artistic expression; a surgical procedure assessment emphasizes precision, safety, and adherence to protocol; a rehabilitation exercise evaluation prioritizes correct movement patterns and pain avoidance.

Traditional approaches often struggle with the multi-modal nature of quality assessment, where expert evaluations typically consider visual form, temporal dynamics, auditory cues, and contextual appropriateness. Real-world videos rarely consist of visual information alone—they incorporate multiple modalities that all contribute to comprehensive understanding. A sports highlight video combines visual action with crowd noise, commentary, and graphical elements that collectively convey the quality and significance of the performance. Expert judges integrate these diverse information sources naturally, while computational approaches have traditionally processed modalities in isolation.

Most current approaches process these modalities in isolation before **late fusion** (combining different modalities only at the final decision stage), missing the complex inter-relationships between different signals that expert human evaluators naturally integrate. Late fusion approaches extract features from each modality independently, then combine them only at the final decision stage. This strategy fails to capture temporal synchronization between visual and audio cues, complementary information across modalities, and contradictory signals that require resolution through cross-modal reasoning.

Our research introduces a novel Mamba-based framework that leverages selective state space models to capture long-range dependencies across multiple information streams. **Mamba** refers to a selective state space model architecture that effectively captures long-

range dependencies across different types of sequential data while maintaining computational efficiency compared to transformer-based approaches [33]. This approach enables effective modeling of the fine-grained temporal patterns and subtle motion nuances that determine action quality, integrating information throughout the assessment process rather than as an afterthought.

1.3 Research Scope and Contributions

This thesis focuses on developing a comprehensive multi-level temporal understanding framework for video analysis that progresses from basic action identification to sophisticated quality evaluation. Rather than exploring the broad spectrum of video analysis tasks, this work concentrates on three core areas that form a natural progression of video comprehension complexity and have significant practical implications across multiple domains.

1.3.1 Research Scope Definition

The research scope encompasses three levels of temporal understanding that collectively enable comprehensive video analysis, as illustrated in Figure 1.2. **Level 1: Action Recognition** addresses the fundamental question of "what" actions occur in video sequences, developing robust methods that focus on genuine motion characteristics rather than contextual biases. **Level 2: Temporal Action Localization** extends beyond recognition to determine precisely "when" actions begin and end in untrimmed videos, requiring integration of contextual relationships and multi-modal information. **Level 3: Action Quality Assessment** evaluates "how well" actions are performed, demanding synthesis of multiple information streams and domain-specific evaluation criteria.

These areas were selected because they address fundamental limitations in current video understanding systems while having clear practical applications across domains including sports analysis, content moderation, healthcare monitoring, and automated assessment systems. The progression from recognition to localization to quality assessment mirrors the development of human expertise and enables increasingly sophisticated video analysis capabilities.

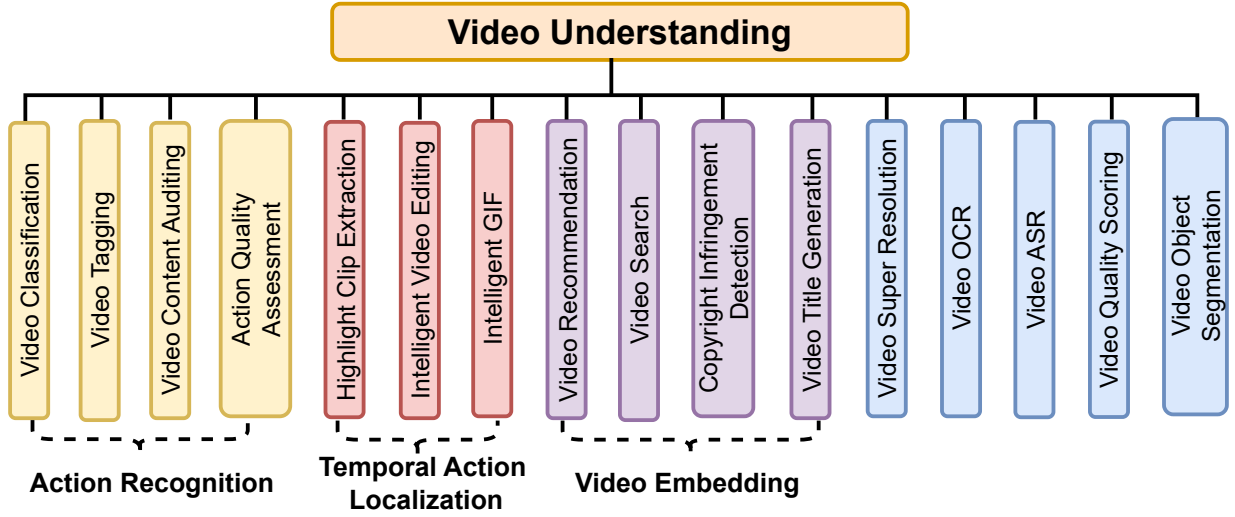


Figure 1.2: Core components and applications of video understanding technology. This figure illustrates the three principal areas: Action Recognition, Temporal Action Localization, and Video Embedding, along with their respective applications in modern video analysis systems.

1.3.2 Primary Research Contributions

This thesis makes three primary contributions corresponding to the three levels of multi-level temporal understanding, collectively advancing both theoretical understanding of video analysis and practical capabilities for real-world applications.

Our action recognition research contributes a hierarchical coarse-to-fine-to-finer framework that addresses motion temporal variability through discriminative feature learning. The approach explicitly contrasts action against non-action segments, fundamentally re-framing action recognition from simple classification to learning motion-critical representations that generalize across domains and execution styles. Through extensive experimentation on benchmark datasets, we demonstrate significant improvements in recognition accuracy while maintaining computational efficiency, particularly in scenarios with domain shifts and varied execution styles. This methodology addresses the motion temporal variability problem where actions manifest with dramatically different patterns across instances, the static bias issue where models preferentially learn from background context rather than action dynamics, and the cross-domain generalization challenge where models trained on one dataset fail to transfer to new environments.

The temporal action localization contribution integrates recognition and localization through comprehensive system design that combines multi-modal fusion, semantic attention mechanisms, and cross-domain prototype alignment. A Class-Semantic Attention mechanism generates precise action proposals by focusing on semantically relevant temporal regions within video sequences. Cross-domain prototype alignment enables knowledge transfer between trimmed and untrimmed video paradigms, addressing the domain shift problem that occurs when models trained on pre-segmented action clips are applied to continuous video streams. By simultaneously addressing both "what" and "when" aspects of action understanding, our system achieves state-of-the-art performance on benchmark datasets and competitive results in international challenges, including second-place rankings in the CVPR ActivityNet Temporal Action Localization competitions.

Our action quality assessment contribution introduces a novel integration strategy for multi-modal information that enables fine-grained evaluation of action execution. Unlike conventional methods that process modalities in isolation before late fusion, our framework integrates information throughout the assessment process, capturing the complex interrelationships between different signals. The framework employs selective attention mechanisms that dynamically focus on relevant temporal segments and modality combinations based on specific assessment criteria. Through experimental validation across domains including sports performance analysis, rehabilitation monitoring, and professional skills assessment, the framework demonstrates superior performance compared to traditional transformer and CNN-based methods in both accuracy and computational efficiency.

1.3.3 Broader Impact and Applications

These three research directions—addressing motion temporal variability in action recognition, contextual relationships in temporal action localization, and multi-modal fusion in action quality assessment—form a cohesive research agenda that advances the broader field of video understanding. While each direction targets a distinct component, they collectively address fundamental challenges in how machines process and interpret dynamic visual content. Our contributions span theoretical advances in modeling temporal information, practical systems for real-world applications, and new frameworks for evaluating action quality.

The impact of our research extends beyond academic advancement to practical applications across numerous domains. Improved action recognition systems enable more effective content moderation and retrieval in large-scale video platforms. Advanced temporal localization capabilities enhance media editing, highlight extraction, and event detection

in surveillance and security systems. Quality assessment frameworks provide new tools for coaching, rehabilitation monitoring, and skills training across sports, healthcare, and professional education. Together, these contributions represent significant steps toward bridging the gap between academic research and practical, deployable video understanding systems that can operate efficiently at scale across diverse real-world scenarios.

1.4 Organization of this Thesis

We first introduce background information and previous research relevant to the subsequent chapters in Chapter 2. This includes a comprehensive literature review of video action recognition methods, temporal action localization techniques, approaches to action quality assessment, contrastive learning frameworks, representation biases in videos, and attention mechanisms in video understanding.

In Chapter 3, we investigate our proposed solution for coarse-to-fine-to-finer discriminative learning on video action recognition. We demonstrate how explicitly modeling the differences between action and non-action segments can create more robust feature representations, and validate our approach through extensive experimental results. This foundational work on feature representation learning establishes the core temporal understanding principles that underpin our subsequent research directions.

In Chapter 4, we build upon our discriminative learning approach and present our comprehensive system for temporal action localization developed for the CVPR ActivityNet Challenge dataset. We detail our approach to video feature extraction, multi-modality classification, and the implementation of attention mechanisms specifically designed for temporal localization tasks. This represents the natural progression from recognizing what actions occur to determining precisely when they occur within the temporal dimension. Experimental results for each component demonstrate the effectiveness of our integrated system and the value of our temporal understanding foundation.

In Chapter 5, we introduce our novel Mamba-based framework for action quality assessment, completing our multi-level temporal understanding paradigm by addressing how well actions are performed. Focusing on effectively evaluating action execution quality through temporal analysis and multi-modal feature integration, this work extends our temporal modeling principles into the qualitative evaluation domain. We demonstrate how our approach captures the nuanced aspects of performance quality across various domains and present extensive experimental validation of our methodology.

In Chapter 6, we conclude this thesis by summarizing our key contributions to video

understanding and discussing future research directions in light of recent advancements in large language models and video-language models. We also reflect on the broader societal impact of our video understanding systems as they continue to find applications across various domains.

This thesis presents a coherent framework for multi-level temporal understanding in videos, progressing systematically from fundamental action recognition through precise temporal localization to nuanced quality assessment. Each chapter builds upon the insights and methodologies of the previous chapters, collectively advancing both the theoretical understanding and practical capabilities in AI-driven video analysis.

Having outlined the structure and objectives of this thesis, we now turn to Chapter 2, where we establish the theoretical and methodological foundations necessary for understanding our research contributions. Through a comprehensive literature review, we will examine the evolution of video understanding techniques and identify persistent challenges that motivate our integrated research agenda. This background will provide essential context for the novel approaches presented in subsequent chapters and situate our work within the broader landscape of artificial intelligence research focused on understanding visual narratives.

Chapter 2

Background

2.1 Introduction

As we embark on our exploration of advanced video understanding techniques, this chapter establishes the theoretical and methodological foundations upon which our research contributions are built. The video-first paradigm described in Chapter 1 presents unique computational challenges that require specialized approaches across multiple dimensions of analysis. This chapter provides a comprehensive review of the state-of-the-art in video understanding, examining the evolution of techniques from traditional hand-crafted features to sophisticated deep learning architectures.

The comprehensive literature review in this chapter is organized around three fundamental challenges that represent critical barriers to achieving human-like video understanding capabilities, as established in Chapter 1. **Challenge 1: Motion Temporal Variability** addresses the phenomenon where identical actions exhibit dramatically different temporal patterns across different instances, contexts, or performers, particularly problematic in action recognition tasks. **Challenge 2: Context-Dependency** concerns the fundamental requirement that video segment interpretation depends critically on global temporal context, essential for accurate temporal action localization in untrimmed videos. **Challenge 3: Multi-Modal Integration** tackles the challenge of combining different information streams (visual, audio, temporal, contextual) to achieve comprehensive understanding, particularly critical in action quality assessment. Each challenge corresponds to a specific level of temporal understanding and represents distinct but foundational elements of the video understanding problem.

We begin with a broad overview of video understanding tasks, situating our specific research areas within the broader landscape of video intelligence. We then trace the evolution of deep learning architectures specifically designed for video data, exploring the computational challenges unique to this domain and the progression from 2D to 3D convolutional paradigms. Our review continues with an examination of temporal modeling approaches, including recurrent neural networks and attention mechanisms, before investigating the transformer revolution that has recently transformed the field. We also explore emerging architectures such as state space models and the Mamba paradigm that promise to further advance video understanding capabilities.

Following this architectural survey, we delve into specific task domains that form the core of our research: action recognition, temporal action localization, and action quality assessment. For each area, we examine foundational concepts, technical evolution, and persistent challenges that motivate our novel contributions in subsequent chapters. By establishing this comprehensive foundation, we create the necessary context for understanding the innovations presented in our research while identifying the limitations in current approaches that our work aims to address.

2.2 Overview of Video Understanding Tasks

To establish a self-contained foundation for this thesis, we first define the key terminology and concepts that underpin our research. **Action recognition** refers to the computational task of identifying and classifying activities occurring in video sequences, typically answering the question "what" action is being performed. **Temporal action localization** extends beyond recognition to determine precisely "when" actions begin and end within continuous video streams, processing **untrimmed videos** (raw footage containing both action segments and non-action background content). **Action quality assessment** evaluates "how well" actions are performed, requiring nuanced evaluation along continuous scales that reflect expert human judgment. **Multi-modal fusion** refers to the integration of different information streams (visual, audio, temporal, contextual) to achieve comprehensive understanding. **Motion temporal variability** describes the phenomenon where identical actions exhibit dramatically different temporal patterns across different instances, contexts, or performers. These foundational concepts form the basis for understanding the three core challenges addressed in this thesis.

The hierarchical decomposition of video understanding tasks constitutes a critical framework for advancing computational models of dynamic scene interpretation [12]. As visualized in Figure 2.1, this taxonomy reflects an evolutionary trajectory from elementary

Overview of Key Video Understanding Tasks

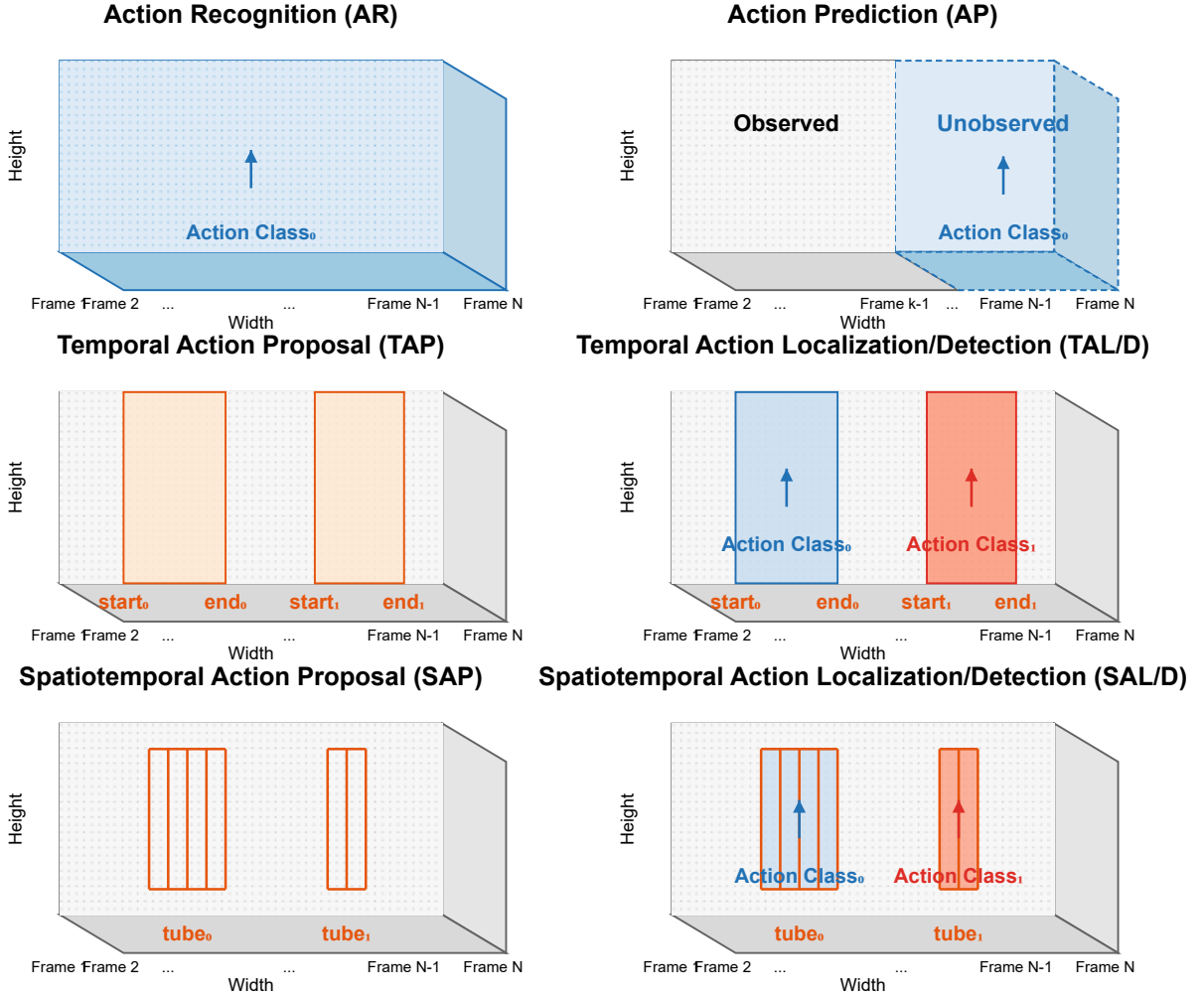


Figure 2.1: A comprehensive overview of key video understanding tasks, encompassing action recognition, action prediction, temporal action proposal, temporal action localization, spatiotemporal action proposal, and spatiotemporal action localization.

pattern recognition to advanced spacetime reasoning – a progression mirroring both the inherent structure of video data and the cognitive processes underlying human visual perception [87]. At its core lies the recognition that video semantics emerge not from isolated frames, but from the intricate interplay of spatial configurations and temporal dynamics

across multiple scales [48].

Action Recognition: The Foundational Layer

Action recognition serves as the cornerstone of video understanding, functioning analogously to object recognition in static image analysis. Over the past decade, this field has witnessed remarkable advancements driven by deep learning innovations. Fundamentally, action recognition divides into two primary categories based on temporal scope and complexity.

Trimmed action recognition focuses on pre-segmented video clips containing single action instances, creating a controlled environment for algorithm development. This approach has evolved from using traditional 3D convolutional neural networks like C3D [93] to more sophisticated architectures such as I3D [12]. These models excel at jointly capturing spatial and temporal patterns through volumetric filters, effectively modeling motion dynamics in standardized datasets like UCF-101 [89]. However, their comprehensive frame-by-frame processing incurs substantial computational costs, presenting scalability challenges.

As research progressed, the limitations of trimmed action recognition became apparent: Real-world videos rarely contain isolated actions. This realization led to the development of untrimmed action recognition, which addresses longer video streams where relevant actions appear as brief segments amidst irrelevant content. This paradigm shift necessitated fundamental architectural innovations to selectively focus on significant temporal segments while efficiently processing extended sequences. The ActivityNet dataset [104] exemplifies this more realistic and challenging scenario.

The methodological progression in action recognition reveals an insightful trajectory. Early approaches decoupled spatial and temporal analysis, treating videos essentially as sequences of 2D frames processed through separate pathways. The introduction of 3D convolutions marked a critical advancement by enabling unified spatiotemporal modeling, though with computational inefficiencies. Subsequent research focused on optimizing this trade-off through factorized architectures like (2+1)D convolutions [95], which decompose 3D operations into more manageable components while preserving modeling capacity.

A crucial conceptual breakthrough emerged with the recognition that human actions possess inherent hierarchical structure—atomic movements combine to form complex activities through temporal composition. This insight inspired temporal pyramid networks and multi-scale transformer architectures that explicitly model action dynamics across varying time scales, capturing both fine-grained motions and broader activity patterns simultaneously.

Contemporary research emphasizes two parallel trends: the value of large-scale pretraining and the growing importance of attention mechanisms. Models pretrained on extensive datasets like Kinetics-400 [49] consistently demonstrate superior generalization capabilities. Concurrently, transformer-based architectures such as TimeSformer [6] have revolutionized untrimmed action recognition through their dynamic attention allocation, though computational complexity remains a concern.

Most recently, the state space model-based Mamba architecture [33] has emerged as a promising alternative for long sequence modeling with linear scaling properties. Early adaptations like "Manta" [40] for few-shot recognition and "Simba" [16] for skeletal action analysis demonstrate Mamba's potential for efficiently processing extended video sequences. While these applications remain in early stages, they suggest a promising direction for addressing the fundamental efficiency-accuracy trade-off that has characterized action recognition research. As this field continues to evolve, integrating hierarchical temporal modeling with efficient sequence processing architectures appears to be the most promising path forward.

Temporal Action Localization: Beyond Classification

Building upon recognition capabilities, temporal action localization (TAL) introduces the critical requirement of temporal boundary precision. This task confronts the dual challenges of proposal generation (identifying candidate action segments) and boundary refinement (precisely localizing start/end frames). Unlike action recognition, TAL must determine both what actions occur and exactly when they transpire within untrimmed videos.

Traditional sliding window approaches proved computationally prohibitive due to the combinatorial explosion of possible temporal segments. The field advanced significantly with boundary-sensitive networks (BSN) [60] that predict action initiation/termination probabilities, enabling efficient proposal generation through temporal boundary regression. BSN employs a "local-to-global" strategy, first identifying high-probability boundary points, then combining these to form proposals, and finally evaluating each proposal's confidence.

Modern graph-based methods like G-TAD [114] further enhance localization accuracy by modeling temporal context through non-local dependencies. This approach represents video snippets as graph nodes with edges indicating snippet correlations, capturing both temporal and semantic contexts. G-TAD employs GCN blocks to learn node features

while dynamically updating edges, allowing the system to resolve ambiguous boundaries by considering adjacent and overlapping actions.

The temporal localization challenge exposes a phenomenon known as *temporal boundary ambiguity*, where even human annotators may disagree on exact start/end points. Recent research employs probabilistic approaches to manage this uncertainty, modeling boundary predictions as Gaussian distributions rather than fixed points. Such methods utilize boundary regression losses like Kullback-Leibler divergence to improve detection performance.

TAL methods are typically evaluated on datasets such as THUMOS-14 [42] and ActivityNet [9], with performance measured through mean Average Precision (mAP) across various temporal Intersection over Union (tIoU) thresholds. As research progresses, integrating probabilistic boundary modeling with context-aware architectures appears most promising for addressing the inherent ambiguity in temporal action localization.

Spatiotemporal Action Localization: The Integration Frontier

The apex of current video understanding capabilities resides in spatiotemporal action localization (STAL), demanding simultaneous precision in spatial detection and temporal tracking. This task transcends mere technical integration of spatial and temporal components, requiring unified representations that preserve spacetime continuity. Traditional approaches cascaded frame-level object detectors with temporal trackers, creating error propagation bottlenecks where detection inaccuracies compound over time, significantly degrading performance in longer videos.

The emergence of tubelet-based architectures represents a paradigm shift, treating actions as 4D spacetime entities rather than sequences of independent detections. Methods like TubeR employ transformer architectures to directly predict spacetime tubes through learned anchor cuboids, effectively bypassing frame-level processing for critical action segments [121]. TubeR introduces tubelet queries and tubelet attention modules that model the dynamic spatiotemporal nature of videos, enhancing model capability particularly for videos containing transitional states or scene changes. Additionally, these advanced methods propose context-aware classification heads and action transition regression heads to further improve action classification and temporal boundary precision.

The computational demands of spatiotemporal localization have driven innovations in efficiency-aware modeling. Lightweight architectures employ spatial-temporal decoupling strategies, processing spatial features at key frames while interpolating temporal dynamics

through motion vectors. This approach proves particularly effective in real-time applications like autonomous navigation systems, where latency constraints prohibit full spacetime convolution over high-resolution inputs [61]. Such methodologies represent an important direction in balancing the trade-off between computational efficiency and spatiotemporal precision in action understanding.

Emerging Dimensions and Future Directions

Contemporary research extends this core taxonomy along several critical axes. Multi-modal fusion architectures now integrate audio spectrograms, inertial sensor data, and textual metadata to overcome visual ambiguities – a capability crucial for parsing social interactions where speech and body language convey complementary meanings. The growing emphasis on self-supervised learning addresses annotation scarcity through pretext tasks like temporal ordering prediction and masked spacetime modeling. Perhaps most significantly, the field is evolving toward *compositional action understanding*, where systems parse videos into hierarchical structures of atomic actions, object interactions, and causal relationships. This progression mirrors human cognitive processes in event comprehension and lays the groundwork for true semantic understanding of dynamic scenes.

The taxonomy presented here serves not merely as a categorical framework, but as a roadmap for advancing video understanding systems. Each hierarchical level introduces new challenges that expose fundamental limitations in current approaches, while simultaneously providing building blocks for more sophisticated reasoning capabilities. As we progress from isolated action classification to holistic spacetime reasoning, the field moves closer to its ultimate goal: artificial visual systems that comprehend dynamic scenes with human-like fluidity and contextual awareness.

2.3 Evolution of Deep Learning Architectures in Video Understanding

2.3.1 The Computational Challenge of Video Understanding

Video understanding presents unique challenges distinct from static image analysis due to its inherent *spatiotemporal dimensionality* and complex motion dynamics [48]. At its core, a video sequence manifests as a 4D tensor $\mathcal{X} \in \mathbb{R}^{T \times H \times W \times 3}$, where T denotes temporal

length, $H \times W$ represents spatial resolution, and 3 corresponds to the RGB color channels. This high-dimensional structure necessitates sophisticated architectures capable of simultaneously modeling *spatial semantics* and *temporal dynamics* [87].

A primary challenge in video understanding stems from temporal redundancy within adjacent frames. Video sequences naturally exhibit high cosine similarity between consecutive frames, expressed as $\text{sim}(\mathbf{x}_t, \mathbf{x}_{t+1}) > 0.9$ in typical scenarios. This redundancy necessitates careful consideration of temporal sampling strategies, which can be formalized through a frame sampling function [106].

$$\mathcal{S}(\mathcal{X}) = \{\mathbf{x}_{\tau_i}\}_{i=1}^N, \quad \tau_i \in \{1, \dots, T\} \quad (2.1)$$

where $\mathcal{S}$ represents the sampling strategy and N denotes the number of selected frames.

The challenge of modeling long-range dependencies becomes particularly acute when considering complex human actions that span extended temporal sequences. For activities in standard benchmarks like Kinetics-400, temporal spans frequently exceed $T \geq 128$ frames. The temporal relationship modeling can be expressed through a hierarchical structure:

$$\mathcal{A} = \bigcup_{k=1}^K \mathcal{A}_k^{\text{scale}}, \quad \text{where } \mathcal{A}_k^{\text{scale}} \in \{\text{atomic actions, mid-level interactions, high-level activities}\} \quad (2.2)$$

The fundamental representation learning problem in video understanding involves mapping the input video tensor to a discriminative feature space. This mapping can be formalized as:

$$\mathbf{F} = \mathcal{F}_\theta(\mathcal{X}), \quad \mathcal{X} \in \mathbb{R}^{T \times H \times W \times 3}, \quad \mathbf{F} \in \mathbb{R}^{T' \times H' \times W' \times C} \quad (2.3)$$

where $\mathcal{F}_\theta$ represents a learnable spatiotemporal feature extractor. The compression ratios $\frac{T}{T'}$ and $\frac{HW}{H'W'}$ reflect crucial design choices in temporal and spatial downsampling strategies.

The optimization of video understanding systems must balance multiple competing objectives, which can be formulated as:

$$\min_{\theta} \mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{compute}} + \lambda_2 \mathcal{L}_{\text{temporal}} + \lambda_3 \mathcal{L}_{\text{scale}} \quad (2.4)$$

subject to computational and memory constraints:

$$\text{subject to: } \begin{cases} \text{FLOPs}(\mathcal{F}_\theta) \leq B_{\text{compute}} \\ \text{Memory}(\mathcal{F}_\theta) \leq B_{\text{memory}} \\ \text{Latency}(\mathcal{F}_\theta) \leq B_{\text{time}} \end{cases} \quad (2.5)$$

At its core, video understanding confronts a fundamental trilemma: the delicate balance between computational efficiency, temporal modeling capacity, and multi-scale feature representation. This three-way trade-off has persistently shaped architectural innovations, driving researchers to seek novel solutions that can optimize across these competing objectives simultaneously.

The historical evolution of video understanding architectures vividly illustrates this ongoing pursuit of balance, as illustrated in Figure 2.2. Early approaches prioritized computational efficiency through frame-wise processing with 2D CNNs, but this came at the cost of limited temporal modeling capacity. The field then witnessed a strategic shift toward 3D CNNs, which enhanced spatiotemporal representation but introduced significant computational overhead. A breakthrough emerged with Transformer-based architectures, which elegantly addressed the temporal modeling challenge through self-attention mechanisms, yet still faced efficiency constraints at scale. Most recently, the introduction of selective state space models (SSMs), exemplified by architectures like Mamba, has opened promising new directions. These modern approaches, particularly when combined with hybrid architectures, suggest a potential path to simultaneously addressing all three aspects of the fundamental trilemma, as depicted in the evolution timeline of Figure 2.2. This architectural progression not only reflects the field’s methodological maturation but also illuminates future research directions in efficient, scalable video understanding.

2.3.2 From 2D to 3D Convolutional Paradigms

The evolution from 2D to 3D convolutional architectures represents a fundamental shift in video understanding, marked by innovative approaches to spatiotemporal modeling. This transition has been characterized by significant architectural breakthroughs, each addressing specific limitations while introducing new capabilities for temporal feature extraction.

The Two-Stream Dilemma

Early efforts in video understanding were dominated by the two-stream architecture introduced by Simonyan & Zisserman [87]. This pioneering approach decomposed video analysis

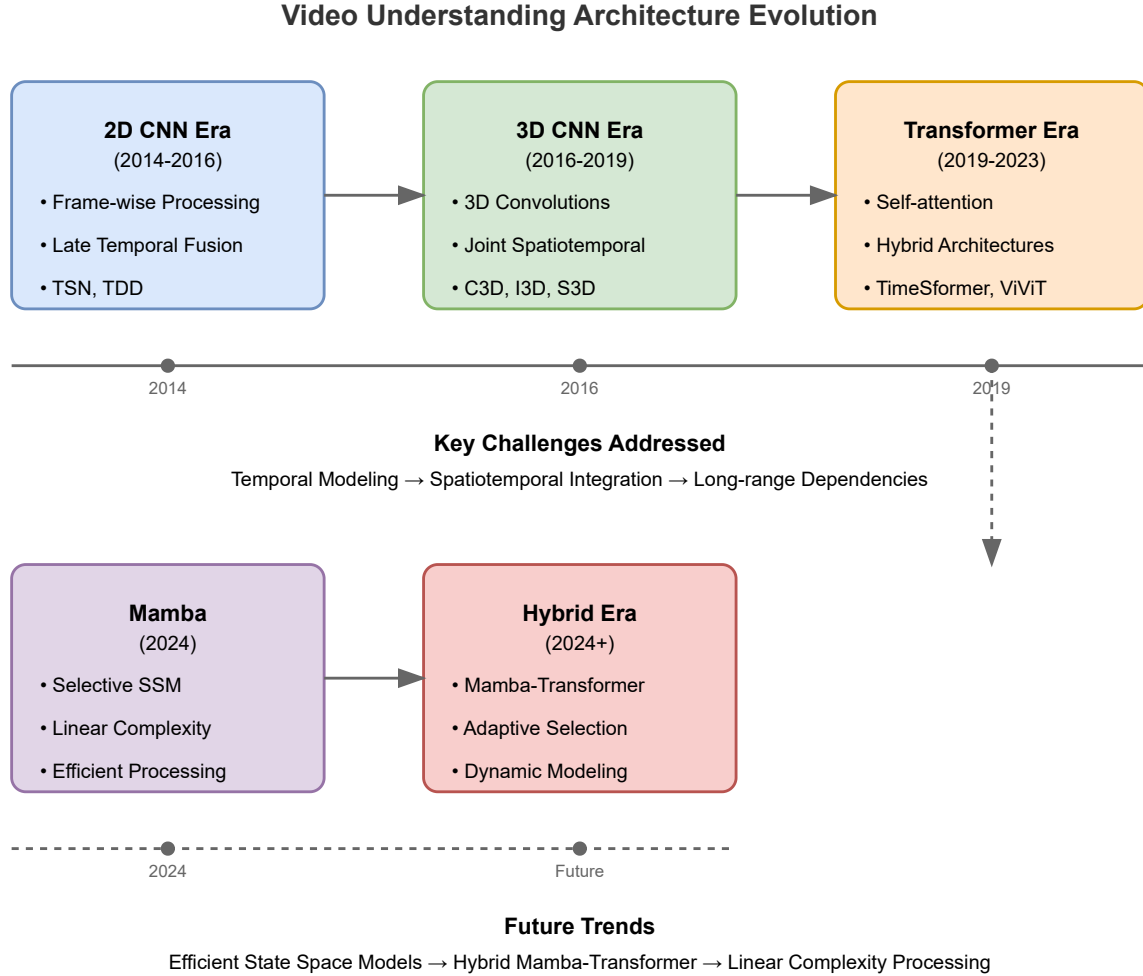


Figure 2.2: Evolution of video understanding architectures from 2014 to present, highlighting three major phases: 2D CNN era (2014-2016) focusing on frame-wise processing, 3D CNN era (2016-2019) introducing joint spatiotemporal modeling, and Transformer era (2019-2023) enabling long-range dependencies. The emergence of Mamba (2024) and hybrid architectures represents a promising future direction towards efficient and adaptive video understanding.

into parallel spatial and temporal pathways, formalized as:

$$\mathbf{y} = \sigma(\mathbf{W}_s \cdot \mathbf{f}_s + \mathbf{W}_t \cdot \mathbf{f}_t + \mathbf{b}) \quad (2.6)$$

where $\mathbf{f}_s \in \mathbb{R}^{d_s}$ and $\mathbf{f}_t \in \mathbb{R}^{d_t}$ represent spatial (RGB) and temporal (optical flow) features, respectively. Despite achieving impressive 88.0% accuracy on UCF101 dataset [89], this approach encountered three critical limitations:

- **Computational Overhead:** The extraction of optical flow using classical methods such as Farnebäck’s algorithm [26] imposed significant computational costs. The optical flow computation can be formalized as:

$$\mathbf{F}_t = \mathcal{O}(\mathbf{I}_t, \mathbf{I}_{t+1}), \quad \text{s.t.} \quad \min_{\mathbf{F}_t} \sum_{x,y} \|\mathbf{I}_{t+1}(x+u, y+v) - \mathbf{I}_t(x, y)\|_2^2 \quad (2.7)$$

where $\mathbf{F}_t$ represents the optical flow field and $\mathbf{I}_t, \mathbf{I}_{t+1}$ are consecutive frames. This preprocessing step consumed a substantial portion of the end-to-end processing time, creating a significant bottleneck in real-time applications.

- **Temporal Fragmentation:** The frame-wise processing strategy with late fusion (typically implemented through softmax normalization) struggled to capture meaningful inter-frame correlations. This limitation can be quantified through the mutual information between temporal features:

$$\mathcal{I}(\mathbf{f}_t^{(i)}, \mathbf{f}_t^{(j)}) = \sum_{x,y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}, \quad \text{for features at frames } i, j \quad (2.8)$$

Empirical analysis revealed significantly degraded feature correlations for frames separated by more than a few timesteps, indicating the model’s inability to capture long-range temporal dependencies effectively.

- **Modality Imbalance:** The learned models exhibited a pronounced bias towards spatial information, quantifiable through the relative contribution of spatial and temporal streams:

$$\lambda = \frac{\|\mathbf{W}_s\|_F^2}{\|\mathbf{W}_t\|_F^2}, \quad \text{where} \quad \|\mathbf{W}\|_F = \sqrt{\sum_{i,j} |w_{ij}|^2} \quad (2.9)$$

This imbalance consistently favored the spatial stream in the decision-making process, suggesting that the temporal information was not being effectively utilized despite its importance for motion understanding. The dominance of spatial features often

led to model decisions being primarily based on appearance rather than motion cues, limiting the architecture’s ability to distinguish between actions with similar spatial contexts but different temporal evolution.

These fundamental limitations of the two-stream approach motivated the development of more integrated architectures that could jointly model spatial and temporal information. The observed challenges particularly highlighted the need for:

$$\mathcal{L}_{\text{total}} = \alpha\mathcal{L}_{\text{spatial}} + \beta\mathcal{L}_{\text{temporal}} + \gamma\mathcal{L}_{\text{fusion}} \quad (2.10)$$

where the balance between spatial ($\mathcal{L}_{\text{spatial}}$), temporal ($\mathcal{L}_{\text{temporal}}$), and fusion ($\mathcal{L}_{\text{fusion}}$) objectives became crucial for subsequent architectural innovations.

3D Convolutional Breakthroughs

The introduction of volumetric convolutions by Ji et al. [46] marked a pivotal advancement in spatiotemporal modeling. As illustrated in Figure 2.3, traditional 2D convolutions process spatial information frame by frame, limiting their ability to capture temporal relationships. To address this limitation, 3D convolutions introduce learnable $t \times h \times w$ kernels that operate directly on video volumes, extending the receptive field into the temporal dimension. This fundamental architectural innovation, visualized in Figure 2.3, enables the network to simultaneously capture both spatial and temporal patterns through unified kernels, representing a significant step forward in integrated spatiotemporal feature learning.

$$\mathbf{F}_{\text{out}}(x, y, z) = \sum_{i=0}^{t-1} \sum_{j=0}^{h-1} \sum_{k=0}^{w-1} \mathcal{K}(i, j, k) \cdot \mathbf{F}_{\text{in}}(x + i, y + j, z + k) \quad (2.11)$$

This novel architecture offered several transformative advantages over the two-stream approach:

- **Joint Spatiotemporal Learning:** The introduction of 3D filters fundamentally changed how temporal information was processed, enabling simultaneous capture of appearance and motion patterns. Unlike the previous two-stream approach where spatial and temporal features were processed independently, 3D convolutions naturally learned the correlations between spatial and temporal dimensions. This unified approach led to more robust temporal feature representations and demonstrated significantly improved capability in capturing complex motion patterns across multiple frames.

2D vs. 3D convolution operations with kernel visualization

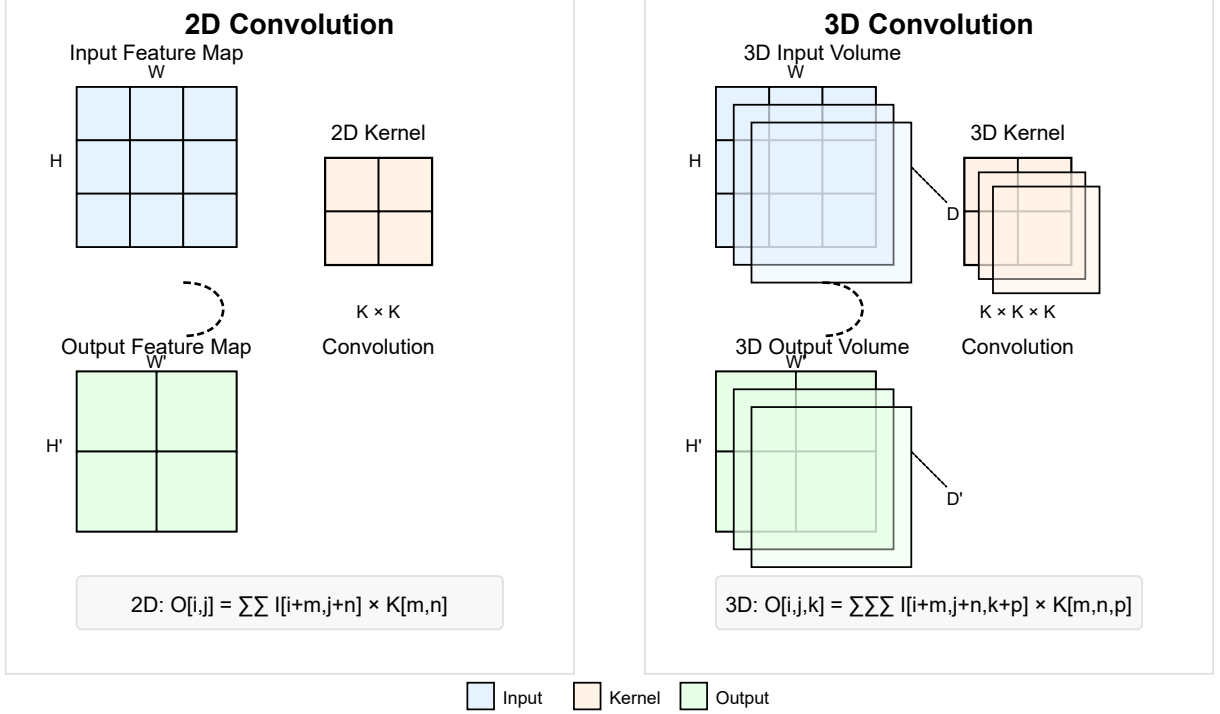


Figure 2.3: 2D vs. 3D convolution operations with kernel visualization. The 3D convolutions operate with $t \times h \times w$ kernels on video volumes, processing temporal and spatial dimensions simultaneously.

- **End-to-End Trainability:** A crucial advancement was the elimination of the optical flow preprocessing stage, which had been a major computational bottleneck in previous approaches. By directly learning from raw video frames, the architecture not only substantially reduced the overall pipeline complexity but also enabled more efficient training and inference processes. This streamlined approach allowed the network to adaptively learn optimal spatiotemporal features directly from the data, rather than relying on hand-crafted motion representations.
- **Hierarchical Abstraction:** The stacked 3D convolutions, exemplified in architectures like C3D [93], introduced a systematic approach to multi-scale feature learning

through a carefully designed progression of layers:

Conv3D-1: $3 \times 3 \times 3$, stride (1, 2, 2)
 Conv3D-2: $3 \times 3 \times 3$, stride (2, 2, 2)
 Conv3D-3: $3 \times 3 \times 3$, stride (2, 2, 2)

This hierarchical structure enabled the network to capture motion patterns at multiple temporal and spatial scales, progressively building more abstract representations. The carefully chosen stride patterns allowed the network to maintain temporal fidelity in early layers while gradually increasing the receptive field, facilitating the recognition of both fine-grained motion details and longer-term action patterns.

These architectural innovations collectively represented a paradigm shift in video understanding, moving away from the separation of spatial and temporal processing towards a more integrated approach. The success of 3D convolutions demonstrated the importance of joint spatiotemporal modeling and set the foundation for subsequent developments in video architecture design.

Architectural Evolution

The development of the C3D architecture [93] marked a decisive turning point in video understanding, representing the first architecture to fully embrace 3D operations throughout its entire network hierarchy. While this approach demonstrated promising results on standard benchmarks, highlighting the potential of pure 3D architectures, it also unveiled two fundamental challenges that would shape subsequent research directions:

- **Parameter Explosion:** The adoption of 3D kernels introduced a significant computational challenge through the cubic scaling of parameters compared to traditional 2D approaches. This relationship can be formally expressed as:

$$\mathcal{P}_{3D} = t \times h \times w \times c_{in} \times c_{out} \quad \text{vs} \quad \mathcal{P}_{2D} = h \times w \times c_{in} \times c_{out} \quad (2.12)$$

This exponential increase in parameter count not only imposed substantial memory requirements but also significantly increased the computational complexity during both training and inference phases.

- **Data Hunger:** The dramatic increase in model complexity introduced a severe data efficiency challenge. The expanded parameter space required substantially more training data than equivalent 2D architectures to achieve proper convergence and prevent overfitting, highlighting the need for more efficient parameter utilization strategies.

These challenges motivated the development of more sophisticated approaches to 3D architectural design. The I3D architecture [12] emerged as a particularly elegant solution through its innovative *network inflation* strategy. Rather than training a 3D network from scratch, I3D proposed a systematic method for leveraging pre-trained 2D CNN knowledge by strategically expanding existing 2D kernels into the temporal dimension:

$$\mathcal{K}_{3D}(i, j, k) = \begin{cases} \mathcal{K}_{2D}(j, k)/t & \text{if } i = \lfloor t/2 \rfloor \\ 0 & \text{otherwise} \end{cases} \quad (2.13)$$

This transformation approach proved particularly powerful as it enabled the network to bootstrap from well-established 2D CNN features while systematically adapting to temporal modeling. The initialization strategy effectively addressed both the parameter efficiency and data hunger challenges by leveraging pre-trained spatial features as a starting point for temporal modeling. This approach not only significantly improved model convergence but also demonstrated substantial performance gains on standard benchmarks compared to randomly initialized networks.

The success of I3D highlighted a crucial insight: while pure 3D architectures offered powerful spatiotemporal modeling capabilities, their effective deployment required careful consideration of parameter efficiency and initialization strategies. This realization would continue to influence subsequent architectural innovations in the field.

Multi-Temporal Modeling

As shown in Figure 2.4, the SlowFast network [28] introduced a revolutionary approach to video understanding by drawing inspiration from biological vision systems, particularly the parallel processing pathways found in the human visual cortex. This architecture recognized that visual signals naturally contain information at different temporal frequencies and devised a dual-stream processing framework to efficiently capture both slow-changing semantic content and rapid motion dynamics. Through its innovative dual-pathway design, the architecture balances computational efficiency with comprehensive temporal modeling:

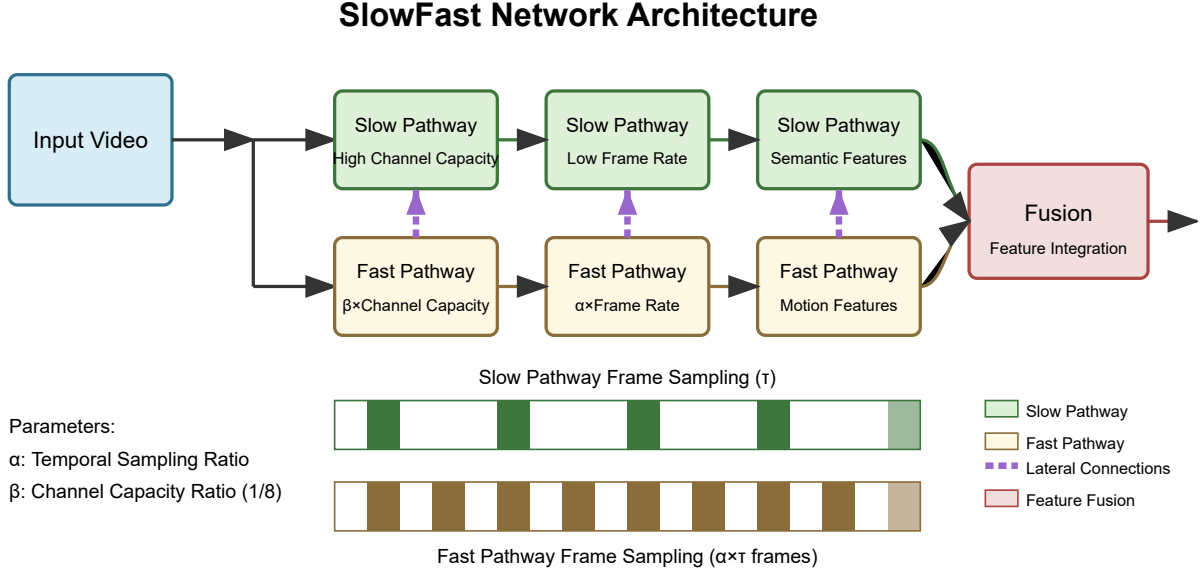


Figure 2.4: The SlowFast Network architecture (adapted from [28]). The network consists of two pathways: a Slow pathway operating at low frame rate for capturing semantic content, and a Fast pathway with $\alpha \times$ higher temporal resolution but reduced channel capacity ($\beta = 1/8$) for efficient motion processing. Lateral connections facilitate information exchange between the pathways.

the Slow pathway processes semantic content at reduced frame rates, while the Fast pathway, despite its higher temporal resolution, maintains efficiency through channel reduction. This biological-inspired division of labor, coupled with strategic lateral connections between pathways, enables effective processing of both semantic and motion information.

The architecture consists of two complementary pathways, each specialized for different aspects of video understanding:

- **Slow Pathway:** This stream operates as the semantic content analyzer, processing sparsely sampled frames with high spatial resolution to capture detailed scene content and gradual temporal evolution:

$$\mathbf{F}_{\text{slow}} = \mathcal{F}_{3D}(\{x_0, x_\tau, x_{2\tau}, \dots\}) \quad (2.14)$$

By focusing on widely spaced temporal samples, this pathway efficiently captures slower-changing scene elements such as object identity, spatial relationships, and

overall context. The high spatial resolution ensures detailed feature extraction from each sampled frame.

- **Fast Pathway:** In parallel, this stream specializes in capturing rapid temporal dynamics through dense temporal sampling but with reduced spatial and channel dimensions:

$$\mathbf{F}_{\text{fast}} = \mathcal{F}_{3D}(\{x_0, x_1, \dots, x_{\alpha\tau}\}), \quad \text{s.t. } \mathcal{C}(\mathbf{F}_{\text{fast}}) = \beta\mathcal{C}(\mathbf{F}_{\text{slow}}) \quad (2.15)$$

This design enables efficient processing of fine-grained motion patterns while maintaining computational efficiency through reduced spatial and channel dimensions. The pathway’s higher temporal resolution allows it to capture subtle motion cues that might be missed by the slow pathway.

The true innovation of the SlowFast architecture lies in its careful integration of these complementary pathways. Through strategically designed lateral connections, the network enables information exchange between the pathways at multiple processing stages. This fusion mechanism allows the model to combine detailed spatial information from the slow pathway with precise motion cues from the fast pathway, creating a rich spatiotemporal representation.

The efficiency of this design stems from its asymmetric allocation of computational resources, dedicating more capacity to spatial processing in the slow pathway while maintaining lightweight but temporally dense processing in the fast pathway. This biological-inspired division of labor proved highly effective, achieving state-of-the-art performance while significantly reducing computational requirements compared to conventional 3D architectures.

This architectural innovation demonstrated a crucial principle in video understanding: the importance of specialized processing streams for different temporal frequencies, mirroring the organization of biological visual systems. The success of the SlowFast network not only advanced the field’s performance benchmarks but also provided valuable insights into the nature of effective temporal modeling in deep neural networks.

2.3.3 Temporal Modeling with Recurrent and Attention Mechanisms

The evolution of temporal modeling in video understanding has witnessed a significant progression from simple recurrent architectures to sophisticated attention mechanisms.

This transition reflects the field’s growing understanding of how to effectively capture and process temporal dependencies at different scales and granularities.

RNNs for Sequential Refinement

Early approaches to temporal modeling centered around hybrid CNN-RNN architectures, with LRCN [21] establishing a foundational framework for sequential processing. Given frame-wise CNN features $\{\mathbf{f}_1, \dots, \mathbf{f}_T\} \in \mathbb{R}^{T \times d}$, where T represents the number of time steps (frames) in the video sequence and d represents the dimensionality of the CNN feature vectors extracted from each frame, the long short-term memory (LSTM) module processes temporal dependencies through:

$$\mathbf{h}_t = \text{LSTM}(\mathbf{f}_t, \mathbf{h}_{t-1}) = \sigma_g(\mathbf{W}_f \mathbf{f}_t + \mathbf{W}_h \mathbf{h}_{t-1} + \mathbf{b}) \quad (2.16)$$

where σ_g denotes the gating mechanisms (sigmoid/tanh) that control information flow, $\mathbf{W}_f \in \mathbb{R}^{d \times d}$ and $\mathbf{W}_h \in \mathbb{R}^{d \times d}$ are the learnable weight matrices for input features and hidden states respectively, and $\mathbf{b} \in \mathbb{R}^d$ is the bias vector. While this approach provided a natural framework for temporal modeling, it revealed several fundamental limitations:

- **Vanishing Gradients:** The recursive nature of RNNs led to exponential decay in gradient propagation across time steps. This phenomenon can be formally characterized through the gradient norm:

$$|\nabla_{\mathbf{h}_0} \mathbf{h}_T| \propto \prod_{t=1}^T |\text{diag}(\sigma'_g(\mathbf{a}_t)) \mathbf{W}_h| \leq (\gamma_{\max})^T \quad (2.17)$$

where $\mathbf{a}_t$ represents the pre-activation input at time step t , σ'_g denotes the derivative of the gating function, and $\gamma_{\max}$ represents the maximum singular value of $\mathbf{W}_h$. This limitation severely constrained the model’s ability to learn long-range dependencies.

- **Temporal Resolution Loss:** The fixed-dimension hidden states $\mathbf{h}_t \in \mathbb{R}^d$ created an information bottleneck, where d denotes the hidden state dimensionality, forcing the network to compress potentially complex temporal dynamics into a fixed-length representation. This compression often resulted in the loss of fine-grained motion details crucial for action recognition.
- **Computational Inefficiency:** The inherently sequential nature of RNN processing prevented parallelization across the temporal dimension, resulting in $\mathcal{O}(T)$ sequential computations for a sequence of length T .

Attention as a Temporal Compressor

The introduction of Temporal Segment Networks (TSN) [101] marked a paradigm shift in temporal modeling through its innovative combination of sparse sampling and attention-based aggregation:

$$\mathbf{F}_{\text{global}} = \sum_{k=1}^K \alpha_k \cdot \mathcal{F}_{\text{CNN}}(\mathcal{X}[t_k]), \quad \alpha_k = \frac{\exp(\mathbf{w}^\top \mathbf{f}_k)}{\sum_{j=1}^K \exp(\mathbf{w}^\top \mathbf{f}_j)} \quad (2.18)$$

where $\mathbf{F}_{\text{global}} \in \mathbb{R}^d$ represents the final global video representation, K is the number of selected segments, α_k denotes the attention weight for the k -th segment, $\mathcal{F}_{\text{CNN}}$ is the CNN feature extraction function, $\mathcal{X}[t_k]$ represents the input frame at time t_k , $\mathbf{w} \in \mathbb{R}^d$ is the learnable attention parameter vector, and $\mathbf{f}_k \in \mathbb{R}^d$ is the feature vector for the k -th segment.

This formulation introduced several key innovations in temporal modeling. The sparse sampling strategy enabled efficient processing of long videos by selecting representative segments, while the attention mechanism allowed the network to learn optimal weighting of these segments based on their relevance to the recognition task.

Spacetime Self-Attention

The development of Non-local Neural Networks [103] represented a significant advancement in temporal modeling by introducing self-attention mechanisms that could capture dependencies across both space and time:

$$\mathbf{F}_{\text{out}} = \text{Softmax} \left(\frac{\mathcal{Q}(\mathbf{F})\mathcal{K}(\mathbf{F})^\top}{\sqrt{d}} \right) \mathcal{V}(\mathbf{F}) + \mathbf{F} \quad (2.19)$$

where $\mathbf{F} \in \mathbb{R}^{H \times W \times T \times d}$ represents the input spacetime feature map with spatial dimensions $H \times W$, temporal dimension T , and feature dimension d . The functions $\mathcal{Q}$, $\mathcal{K}$, and $\mathcal{V}$ denote the query, key, and value transformation functions respectively, each mapping from $\mathbb{R}^d$ to $\mathbb{R}^{d'}$ where d' is the transformed feature dimension, and $\mathbf{F}_{\text{out}} \in \mathbb{R}^{H \times W \times T \times d}$ is the output feature map with residual connection.

This was further enhanced through position-aware attention:

$$\mathbf{F}_{\text{out}}(x, y, t) = \sum_{x', y', t'} \theta(\mathbf{F}(x, y, t), \mathbf{F}(x', y', t')) \cdot \phi(\mathbf{F}(x', y', t')) + \mathbf{F}(x, y, t) \quad (2.20)$$

where (x, y, t) and (x', y', t') represent spatial-temporal coordinates in the feature map, $\theta(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ computes similarity scores between feature vectors at different spacetime positions, $\phi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^d$ generates position-aware value representations, and the summation is performed over all possible spatial-temporal positions (x', y', t') in the feature map.

This progression in attention mechanisms reflects the field’s ongoing efforts to balance computational efficiency with modeling capacity, particularly in the context of long-form video understanding. The development of hybrid approaches suggests a promising direction for future research in scalable temporal modeling.

2.3.4 Transformer Revolution in Video Understanding

The emergence of Transformer architectures has fundamentally reshaped the landscape of computer vision and video understanding. Before delving into video-specific architectures, it is essential to understand the foundations of Transformers and their adaptation to vision tasks through Vision Transformers (ViT) [22].

Transformer Foundations

The Transformer architecture, originally proposed for natural language processing tasks [96], introduces self-attention as its core computational mechanism. The architecture processes a sequence of tokens through alternating layers of Multi-Head Self-Attention (MSA) and Feed-Forward Networks (FFN), as illustrated in Figure 2.5.

The fundamental self-attention mechanism computes relationships between all positions in a sequence simultaneously:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right) \mathbf{V} \quad (2.21)$$

where $\mathbf{Q} \in \mathbb{R}^{N \times d_k}$ represents the query matrix containing N query vectors of dimension d_k , $\mathbf{K} \in \mathbb{R}^{N \times d_k}$ represents the key matrix with corresponding key vectors, $\mathbf{V} \in \mathbb{R}^{N \times d_v}$

Transformer Architecture

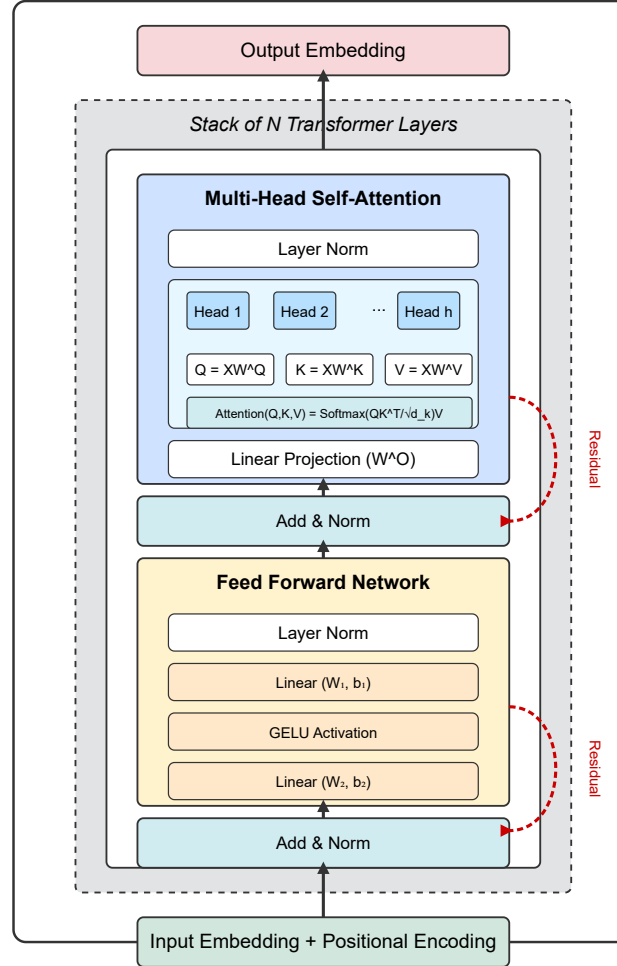


Figure 2.5: Transformer architecture originally proposed for natural language processing tasks. The model processes sequences through alternating layers of Multi-Head Self-Attention (MSA) and Feed-Forward Networks (FFN). In the attention mechanism, queries, keys, and values interact through softmax-based calculations scaled by the dimension of key vectors. The multi-head attention projects inputs into multiple subspaces, with each head performing independent attention operations. Residual connections (shown as dashed red lines) and layer normalization are applied throughout the architecture, creating two sequential processing blocks within each transformer layer.

represents the value matrix containing value vectors of dimension d_v , N is the sequence length (number of input tokens), and $\sqrt{d_k}$ serves as a scaling factor to prevent extremely large values in the attention weights that could lead to vanishing gradients in the softmax function.

The multi-head attention mechanism extends this by projecting the inputs into multiple subspaces, enabling the model to attend to different types of relationships simultaneously:

$$\text{MultiHead}(\mathbf{X}) = [\text{head}_1; \dots; \text{head}_h] \mathbf{W}^O \quad (2.22)$$

$$\text{head}_i = \text{Attention}(\mathbf{X}\mathbf{W}_i^Q, \mathbf{X}\mathbf{W}_i^K, \mathbf{X}\mathbf{W}_i^V) \quad (2.23)$$

where $\mathbf{X} \in \mathbb{R}^{N \times d}$ represents the input feature matrix with d being the input feature dimension, h denotes the number of attention heads, $\mathbf{W}_i^Q, \mathbf{W}_i^K \in \mathbb{R}^{d \times d_k}$ and $\mathbf{W}_i^V \in \mathbb{R}^{d \times d_v}$ are the learnable projection matrices for the i -th head that transform input features into query, key, and value representations respectively, $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d}$ is the output projection matrix that combines information from all heads, and $[\cdot; \cdot]$ denotes concatenation operation.

The complete transformer layer combines self-attention with a position-wise feed-forward network through residual connections and layer normalization:

$$\begin{aligned} \mathbf{z}'_l &= \text{MSA}(\text{LN}(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1} \\ \mathbf{z}_l &= \text{FFN}(\text{LN}(\mathbf{z}'_l)) + \mathbf{z}'_l \end{aligned} \quad (2.24)$$

where $\mathbf{z}_{l-1} \in \mathbb{R}^{N \times d}$ represents the input to the l -th transformer layer, $\mathbf{z}'_l \in \mathbb{R}^{N \times d}$ is the intermediate output after the attention sublayer, $\mathbf{z}_l \in \mathbb{R}^{N \times d}$ is the final output of the l -th layer, $\text{LN}(\cdot)$ denotes Layer Normalization which normalizes features across the feature dimension to stabilize training, and $\text{MSA}(\cdot)$ represents the Multi-Head Self-Attention operation. The feed-forward network consists of two linear transformations with a non-linearity:

$$\text{FFN}(\mathbf{x}) = \text{GELU}(\mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (2.25)$$

where $\mathbf{x} \in \mathbb{R}^d$ represents the input vector, $\mathbf{W}_1 \in \mathbb{R}^{d \times d_{\text{ff}}}$ and $\mathbf{W}_2 \in \mathbb{R}^{d_{\text{ff}} \times d}$ are the weight matrices of the two linear layers, $\mathbf{b}_1 \in \mathbb{R}^{d_{\text{ff}}}$ and $\mathbf{b}_2 \in \mathbb{R}^d$ are the corresponding bias vectors, d_{ff} represents the inner dimension of the feed-forward network (typically $4d$), and $\text{GELU}(\cdot)$ denotes the Gaussian Error Linear Unit activation function that provides smooth, non-linear transformations.

Attention as a Temporal Compressor

The introduction of Temporal Segment Networks (TSN) [101] marked a paradigm shift in temporal modeling through its innovative combination of sparse sampling and attention-based aggregation:

$$\mathbf{F}_{\text{global}} = \sum_{k=1}^K \alpha_k \cdot \mathcal{F}_{\text{CNN}}(\mathcal{X}[t_k]), \quad \alpha_k = \frac{\exp(\mathbf{w}^\top \mathbf{f}_k)}{\sum_{j=1}^K \exp(\mathbf{w}^\top \mathbf{f}_j)} \quad (2.26)$$

where $\mathbf{F}_{\text{global}} \in \mathbb{R}^d$ represents the final global video representation, K is the number of selected segments, α_k denotes the attention weight for the k -th segment, $\mathcal{F}_{\text{CNN}}$ is the CNN feature extraction function, $\mathcal{X}[t_k]$ represents the input frame at time t_k , $\mathbf{w} \in \mathbb{R}^d$ is the learnable attention parameter vector, and $\mathbf{f}_k \in \mathbb{R}^d$ is the feature vector for the k -th segment.

This formulation introduced several key innovations in temporal modeling. The sparse sampling strategy enabled efficient processing of long videos by selecting representative segments, while the attention mechanism allowed the network to learn optimal weighting of these segments based on their relevance to the recognition task.

Spacetime Self-Attention

The development of Non-local Neural Networks [103] represented a significant advancement in temporal modeling by introducing self-attention mechanisms that could capture dependencies across both space and time:

$$\mathbf{F}_{\text{out}} = \text{Softmax} \left(\frac{\mathcal{Q}(\mathbf{F})\mathcal{K}(\mathbf{F})^\top}{\sqrt{d}} \right) \mathcal{V}(\mathbf{F}) + \mathbf{F} \quad (2.27)$$

where $\mathbf{F} \in \mathbb{R}^{H \times W \times T \times d}$ represents the input spacetime feature map with spatial dimensions $H \times W$, temporal dimension T , and feature dimension d . The functions $\mathcal{Q}$, $\mathcal{K}$, and $\mathcal{V}$ denote the query, key, and value transformation functions respectively, each mapping from $\mathbb{R}^d$ to $\mathbb{R}^{d'}$ where d' is the transformed feature dimension, and $\mathbf{F}_{\text{out}} \in \mathbb{R}^{H \times W \times T \times d}$ is the output feature map with residual connection.

This was further enhanced through position-aware attention:

$$\mathbf{F}_{\text{out}}(x, y, t) = \sum_{x', y', t'} \theta(\mathbf{F}(x, y, t), \mathbf{F}(x', y', t')) \cdot \phi(\mathbf{F}(x', y', t')) + \mathbf{F}(x, y, t) \quad (2.28)$$

where (x, y, t) and (x', y', t') represent spatial-temporal coordinates in the feature map, $\theta(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ computes similarity scores between feature vectors at different spacetime positions, $\phi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^d$ generates position-aware value representations, and the summation is performed over all possible spatial-temporal positions (x', y', t') in the feature map.

From Transformers to Vision Transformers

The adaptation of Transformers to vision tasks required several key innovations to handle the structured nature of visual data. Vision Transformer (ViT) [22] introduced these critical adaptations:

1) **Image Tokenization**: Unlike text tokens in NLP, images are divided into fixed-size patches to create a sequence that transformers can process:

$$\mathbf{x}_p = [\text{patch}_1, \text{patch}_2, \dots, \text{patch}_N], \quad \text{patch}_i \in \mathbb{R}^{P^2 \cdot C} \quad (2.29)$$

where $\mathbf{x}_p$ represents the sequence of image patches created by dividing the original image into non-overlapping square regions, N is the total number of patches obtained from the image, each patch of size $P \times P$ pixels is flattened into a one-dimensional vector of length $P^2 \cdot C$, and C represents the number of input channels (e.g., $C = 3$ for RGB color images, $C = 1$ for grayscale images). This tokenization process transforms the two-dimensional image structure into a sequential format that can be processed by transformer attention mechanisms. Each flattened patch is then projected into a D -dimensional embedding space through a learnable linear transformation:

$$\mathbf{E} = [\mathbf{x}_p^1 \mathbf{W}_E; \mathbf{x}_p^2 \mathbf{W}_E; \dots; \mathbf{x}_p^N \mathbf{W}_E], \quad \mathbf{W}_E \in \mathbb{R}^{(P^2 \cdot C) \times D} \quad (2.30)$$

where $\mathbf{E} \in \mathbb{R}^{N \times D}$ represents the patch embedding matrix containing the transformed representations of all image patches, $\mathbf{W}_E$ is the learnable linear projection matrix that maps each flattened patch vector to a D -dimensional embedding, and D is the embedding dimension used consistently throughout the transformer architecture. This projection allows patches of varying visual content to be represented in a unified high-dimensional space suitable for attention computations.

2) **Position Encoding:** To preserve spatial information lost during the flattening and tokenization process, learnable position embeddings are added to encode the original spatial locations of each patch:

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{E}] + \mathbf{E}_{\text{pos}}, \quad \mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1) \times D} \quad (2.31)$$

where $\mathbf{z}_0 \in \mathbb{R}^{(N+1) \times D}$ represents the complete input to the first transformer layer, $\mathbf{x}_{\text{class}} \in \mathbb{R}^{1 \times D}$ is a learnable classification token that serves as a global representation for the entire image and is used for final classification, $\mathbf{E}_{\text{pos}}$ contains learnable position embeddings that encode the spatial coordinates of all N patches plus the classification token, and $[\cdot; \cdot]$ denotes concatenation along the sequence dimension. The position embeddings enable the model to understand which patches are spatially adjacent and maintain awareness of the original image structure despite the sequential processing format.

3) **Hierarchical Feature Learning:** While original Transformers process sequences at a single resolution scale, vision tasks often require multi-scale feature representations to capture both fine-grained details and global context:

$$\mathbf{F}_l = \text{Transformer}_l(\mathbf{F}_{l-1}), \quad l = 1 \dots L \quad (2.32)$$

where $\mathbf{F}_l \in \mathbb{R}^{(N+1) \times D}$ represents the feature representation at layer l , Transformer_l denotes the l -th transformer block consisting of multi-head self-attention followed by a feed-forward network with residual connections and layer normalization, L is the total number of transformer layers in the architecture, and each layer l progressively refines the feature representations through attention mechanisms that can capture increasingly complex visual patterns and relationships between patches.

The success of these adaptations rests on several key insights that address fundamental differences between language and vision processing:

- **Local-Global Processing:** Self-attention naturally enables both local pattern recognition (relationships between nearby patches) and global context understanding (relationships across the entire image) without requiring explicit convolution operations or predefined receptive fields, allowing the model to adaptively determine which spatial relationships are most important for the task.
- **Position Information:** Unlike language where word order provides inherent positional meaning, images require explicit encoding of 2D spatial relationships to understand how patches relate to each other in the original image coordinate system, which

is addressed through learnable position embeddings that capture both horizontal and vertical spatial relationships.

- **Scale Handling:** Vision tasks often require processing visual information at multiple scales simultaneously—from fine details within individual patches to global patterns across the entire image—necessitating architectural adaptations that enable transformers to effectively handle this multi-scale nature of visual understanding.

The computational complexity of Vision Transformers can be expressed as:

$$\mathcal{C}_{\text{ViT}} = \underbrace{HW \cdot P^2 C \cdot D}_{\text{patch embedding}} + \underbrace{L \cdot (N^2 \cdot D + N \cdot D^2)}_{\text{transformer layers}} \quad (2.33)$$

where H and W are the input image height and width dimensions respectively (measured in pixels), P is the patch size representing both the height and width of each square patch, C is the number of input channels in the image, D is the embedding dimension used throughout the network architecture, L is the number of transformer layers, and $N = (H \cdot W)/P^2$ is the total number of patches obtained by dividing the image into non-overlapping square regions. The first term represents the computational cost of converting image patches into embeddings, while the second term captures the cost of processing these embeddings through the transformer layers, where $N^2 \cdot D$ corresponds to the attention computation complexity and $N \cdot D^2$ represents the feed-forward network complexity.

Following the success of Vision Transformers in image understanding, extending transformer architectures to the video domain became a natural yet challenging progression. Video understanding presents unique challenges beyond static image processing, requiring models to capture not only spatial relationships within individual frames but also complex temporal dynamics and motion patterns across multiple frames in a sequence. This temporal dimension adds significant complexity as models must learn to distinguish between relevant motion patterns and irrelevant variations while maintaining computational efficiency for processing long video sequences. This section examines how transformer architectures evolved to address these multifaceted challenges, marking a revolutionary turning point in video understanding capabilities.

Vanilla Spacetime Attention

The first crucial step in this evolution came with TimeSformer [6], which pioneered the adaptation of Vision Transformers to handle spatiotemporal data by extending the concept

of image patch processing to video sequences. While ViT processed 2D image patches from static images, TimeSformer extended this concept to 3D video volumes by treating video as a collection of patches across both spatial and temporal dimensions.

Given an input video segmented into patches $\mathbf{P} \in \mathbb{R}^{N \times T \times d}$, where $N = H' \times W'$ represents the total number of spatial patches per frame (with H' and W' being the number of patches along height and width dimensions respectively), T denotes the temporal depth (number of frames in the video sequence), and d represents the feature dimension of each patch embedding, the core mechanism of spacetime self-attention is formulated as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right) \mathbf{V} \quad (2.34)$$

Here, $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N_T \times d}$ represent query, key, and value matrices respectively for $N_T = N \times T$ total spacetime tokens (where each token corresponds to a patch at a specific spatial location and temporal position), and $\sqrt{d}$ serves as a scaling factor to normalize attention weights and prevent gradient vanishing issues.

The TimeSformer architecture explored three distinct attention variants, each offering different trade-offs between computational efficiency and modeling capacity for capturing spatiotemporal relationships:

- **Space-only Attention:** This variant processes spatial relationships within each frame independently, treating each temporal frame separately:

$$\mathbf{A}_t = \text{Softmax}\left(\frac{\mathbf{Q}_t \mathbf{K}_t^\top}{\sqrt{d}}\right), \quad t \in \{1, \dots, T\} \quad (2.35)$$

where $\mathbf{Q}_t, \mathbf{K}_t \in \mathbb{R}^{N \times d}$ represent queries and keys for all spatial patches at time t , and $\mathbf{A}_t \in \mathbb{R}^{N \times N}$ is the attention matrix capturing spatial relationships within frame t . This approach maintains computational efficiency with $\mathcal{O}(TN^2d)$ complexity but limits temporal modeling capacity since it cannot capture motion or temporal dependencies between frames.

- **Time-only Attention:** This mechanism focuses on temporal relationships at each spatial position, computing attention across time for patches at the same spatial coordinates:

$$\mathbf{A}_{xy} = \text{Softmax}\left(\frac{\mathbf{Q}_{xy} \mathbf{K}_{xy}^\top}{\sqrt{d}}\right), \quad (x, y) \in \{1, \dots, H'\} \times \{1, \dots, W'\} \quad (2.36)$$

where $\mathbf{Q}_{xy}, \mathbf{K}_{xy} \in \mathbb{R}^{T \times d}$ represent queries and keys for the patch at spatial position (x, y) across all T time steps, and $\mathbf{A}_{xy} \in \mathbb{R}^{T \times T}$ captures how the same spatial region evolves over time. This design captures temporal dynamics while maintaining reasonable computational requirements of $\mathcal{O}(NT^2d)$, making it effective for modeling local motion patterns.

- **Joint Spacetime Attention:** This most comprehensive variant enables full attention across all spatiotemporal tokens, allowing each patch at any spatial and temporal position to attend to every other patch in the video:

$$\mathbf{A} = \text{Softmax} \left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}} \right) \in \mathbb{R}^{N_T \times N_T} \quad (2.37)$$

where $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{N_T \times d}$ contain queries and keys for all $N_T = N \times T$ spacetime tokens, and $\mathbf{A}$ represents the complete attention matrix capturing all possible spatiotemporal relationships. While computationally intensive with $\mathcal{O}((NT)^2d)$ complexity, this approach offers the most complete modeling of spatiotemporal relationships, enabling the capture of complex motion patterns, long-range temporal dependencies, and global spatiotemporal context.

The choice between these attention variants represents a fundamental trade-off in video transformer design: space-only attention provides computational efficiency but sacrifices temporal modeling, time-only attention captures temporal dynamics efficiently but may miss spatial context, while joint spacetime attention provides comprehensive modeling at the cost of quadratic computational complexity in the number of spacetime tokens.

Hierarchical Transformers

To address the computational challenges of full attention mechanisms, the Video Swin Transformer [67] introduced a hierarchical processing framework with shifted windows that dramatically reduces computational complexity while maintaining effective spatiotemporal modeling capabilities. This innovative approach fundamentally changes how attention is computed by restricting it to local windows rather than global tokens.

The core window-based attention mechanism can be formalized as:

$$\text{WindowAttention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{LocalSoftmax} \left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}} \right) \mathbf{V} \quad (2.38)$$

where $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{k^3 \times d}$ represent query, key, and value matrices within each local $k \times k \times k$ spatiotemporal window (where k is the window size in each dimension), d is the feature dimension, and LocalSoftmax computes attention weights only among tokens within the same window, significantly reducing computational requirements from quadratic global attention to linear local attention.

The architecture employs non-overlapping $k \times k \times k$ windows that partition the video volume into smaller, manageable regions. To enable information flow between these isolated windows, a shifted window mechanism is introduced:

$$\text{Shift}(x, y, t) = (x + \lfloor k/2 \rfloor \bmod k, y + \lfloor k/2 \rfloor \bmod k, t + \lfloor k/2 \rfloor \bmod k) \quad (2.39)$$

where (x, y, t) represents the original spatial-temporal coordinates of a patch, $\lfloor k/2 \rfloor$ is the shift amount (half the window size), and the modulo operation ensures the shift wraps around window boundaries. This shifting mechanism alternates between layers, ensuring that regions split by window boundaries in one layer can interact in the subsequent layer, enabling cross-window communication while maintaining computational efficiency.

This window-based design leads to a significant reduction in computational complexity compared to global attention:

$$\mathcal{C}_{\text{global}} = \mathcal{O}(T^2 H W D) \xrightarrow{\text{window}} \mathcal{C}_{\text{window}} = \mathcal{O}(T H W k^3 D) \quad (2.40)$$

where T is the number of temporal frames, H and W are spatial dimensions, D is the feature dimension, and k is the window size. The global attention complexity is quadratic in the total number of tokens, while the window-based approach achieves linear complexity by limiting attention computation to local neighborhoods of size k^3 .

Multi-Scale Architectural Design

The hierarchical framework implements a progressive downsampling strategy through four distinct stages, each operating at different spatial and temporal resolutions to capture features at multiple scales:

$$\begin{aligned}
\text{Stage 1:} \quad & \text{Conv3D } 7 \times 7 \times 7, \text{ stride } (2, 4, 4) \rightarrow T \times \frac{H}{4} \times \frac{W}{4} \\
\text{Stage 2:} \quad & \text{Swin Block (k=4)} \rightarrow \frac{T}{2} \times \frac{H}{8} \times \frac{W}{8} \\
\text{Stage 3:} \quad & \text{Swin Block (k=8)} \rightarrow \frac{T}{4} \times \frac{H}{16} \times \frac{W}{16} \\
\text{Stage 4:} \quad & \text{Swin Block (k=16)} \rightarrow \frac{T}{8} \times \frac{H}{32} \times \frac{W}{32}
\end{aligned} \tag{2.41}$$

Here, Stage 1 employs a 3D convolution with kernel size $7 \times 7 \times 7$ and stride $(2, 4, 4)$ for initial feature extraction and dimensionality reduction, where the stride values indicate downsampling by factor 2 in time and factor 4 in both spatial dimensions. Stages 2-4 use Swin Blocks with progressively increasing window sizes $k \in \{4, 8, 16\}$, where larger windows at later stages capture broader spatial and temporal context as the feature maps become smaller. The output dimensions show progressive downsampling: temporal resolution is halved at each stage after Stage 1, while spatial resolution is consistently reduced by factor 2 at each stage.

Each Swin Block incorporates both attention and feed-forward components with residual connections:

$$\mathbf{F}' = \text{WindowAttention}(\text{LN}(\mathbf{F})) + \mathbf{F}, \quad \mathbf{F}'' = \text{MLP}(\text{LN}(\mathbf{F}')) + \mathbf{F}' \tag{2.42}$$

where $\mathbf{F} \in \mathbb{R}^{N_w \times k^3 \times d}$ represents the input features organized into N_w windows each containing k^3 tokens of dimension d , $\text{LN}(\cdot)$ denotes Layer Normalization applied before each sub-layer, $\text{WindowAttention}(\cdot)$ performs the local attention computation within each window, $\text{MLP}(\cdot)$ represents a multi-layer perceptron (feed-forward network) typically consisting of two linear layers with GELU activation, and the residual connections (+) help stabilize training and enable gradient flow through deep networks. The prime notation indicates intermediate outputs: $\mathbf{F}'$ after the attention sublayer and $\mathbf{F}''$ as the final output of the Swin Block.

This hierarchical design enables the model to process fine-grained details at early stages with small windows while capturing global context at later stages with larger windows, effectively balancing computational efficiency with comprehensive spatiotemporal understanding.

Computational Complexity Analysis

For comprehensive understanding, we analyze the computational requirements of different attention mechanisms:

- **Global Attention:**

$$\mathcal{C}_{\text{global}} = 4THWd^2 + 2(THW)^2d \quad (2.43)$$

- **Window Attention:**

$$\mathcal{C}_{\text{window}} = 4THWd^2 + 2THWk^3d \quad (2.44)$$

- **Shifted Window Mechanism:**

$$\mathcal{C}_{\text{shift}} = \mathcal{C}_{\text{window}} + \frac{THW}{k^3} \cdot 9k^3d = \mathcal{C}_{\text{window}} + 9THWd \quad (2.45)$$

Position Encoding Strategies

The effectiveness of Transformer architectures in video understanding heavily relies on appropriate position encoding mechanisms. Unlike vanilla transformers that only need to encode 1D sequence positions, video transformers must encode both spatial locations within frames and temporal positions across frames. The spatiotemporal position embeddings are designed to capture this dual nature:

$$\mathbf{E}_{\text{pos}} = \{\mathbf{e}_{i,j,t}\}_{i,j,t} \in \mathbb{R}^{H \times W \times T \times D} \quad (2.46)$$

where each position embedding $\mathbf{e}_{i,j,t}$ is decomposed into spatial and temporal components:

$$\mathbf{e}_{i,j,t} = \mathbf{E}_{\text{spatial}}(i, j) + \mathbf{E}_{\text{temporal}}(t) \quad (2.47)$$

Here, $\mathbf{E}_{\text{spatial}} : \mathbb{R}^2 \rightarrow \mathbb{R}^D$ maps 2D spatial coordinates (i, j) to a D -dimensional embedding space, and $\mathbf{E}_{\text{temporal}} : \mathbb{R} \rightarrow \mathbb{R}^D$ maps temporal position t to the same embedding space. This additive decomposition enables the model to learn separate spatial and temporal patterns while maintaining their interactions. The approach proves particularly effective as it allows the transformer to preserve spatial relationships within individual frames while simultaneously capturing temporal dependencies across the video sequence. By sharing

position information across different channels in the embedding space, the model can develop rich representations that integrate both spatial and temporal contexts, essential for understanding complex video dynamics. The resulting position encoding thus provides the transformer with a comprehensive framework for learning spatiotemporal patterns, fundamentally supporting its ability to process and understand video content.

The sophisticated integration of position encoding with attention mechanisms represents a key milestone in transformer-based video understanding. While position encoding provides the essential spatiotemporal context, the attention mechanisms leverage this information to selectively focus on relevant features across both space and time. Together, these complementary components form the foundation of modern video transformers.

This architectural evolution in Transformer-based video understanding demonstrates the field’s progression toward more efficient and effective spatiotemporal modeling. The development of hierarchical and windowed attention mechanisms represents a crucial step in balancing computational efficiency with modeling capacity, while position encoding strategies ensure the preservation of crucial spatiotemporal relationships. These advances have established a robust foundation for future research in video understanding, particularly in addressing the challenges of long-form video analysis and real-time processing requirements.

2.3.5 State Space Models and the Mamba Paradigm

The evolution of video understanding architectures has recently witnessed a significant paradigm shift with the emergence of state space models (SSMs), particularly through the Mamba architecture [33]. Mamba introduces a state space model architecture that offers several compelling advantages for video processing, fundamentally addressing the limitations of both convolutional and attention-based approaches. This novel approach provides an efficient mechanism for capturing long-range dependencies while maintaining linear computational complexity with respect to sequence length, making it particularly valuable for video understanding where processing long sequences has traditionally been computationally challenging.

Mathematical Foundation of Selective SSMs

At its core, Mamba introduces a revolutionary input-dependent state transition mechanism that enables selective processing of temporal information. Given a video sequence $\mathbf{X} \in$

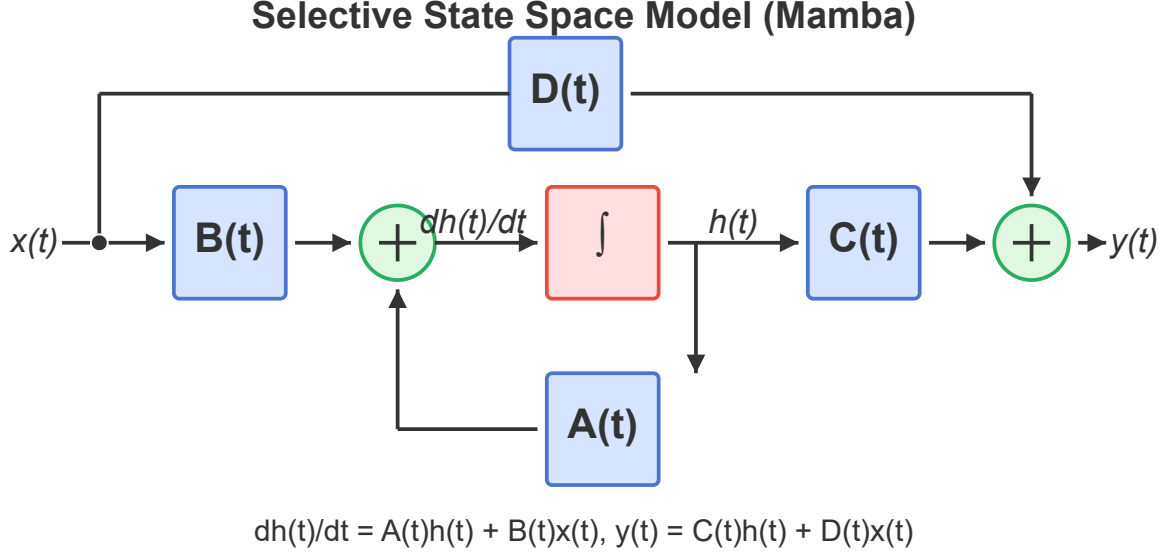


Figure 2.6: Selective state space model (SSM) architecture inspired by Mamba [33]. The system receives an input sequence $\mathbf{x}(t)$, dynamically generates a state transition matrix $\mathbf{A}(t)$ via an MLP, and discretizes the continuous-time system using a Zero-Order Hold (ZOH) mechanism. The hidden state $\mathbf{h}(t)$ evolves based on $\mathbf{A}(t)$ and $\mathbf{B}(t)$, and the output $\mathbf{y}(t)$ is computed through $\mathbf{C}(t)$ and $\mathbf{D}(t)$.

$\mathbb{R}^{T \times D}$ with T timesteps and D -dimensional features, the selective state space model is formulated through a continuous-time state evolution system:

$$\begin{aligned} \frac{d\mathbf{h}(t)}{dt} &= \mathbf{A}(t)\mathbf{h}(t) + \mathbf{B}(t)\mathbf{x}(t) \\ \mathbf{y}(t) &= \mathbf{C}(t)\mathbf{h}(t) + \mathbf{D}(t)\mathbf{x}(t) \end{aligned} \tag{2.48}$$

This formulation is illustrated in Figure 2.6, where each module corresponds to a component in the structured state space model, and the input-dependent design enables selective integration of temporal information through parameterized data-dependent state transitions.

The key innovation lies in making the state transition matrix $\mathbf{A}(t) \in \mathbb{R}^{N \times N}$ input-dependent through parameterization:

$$\mathbf{A}(t) = \text{MLP}(\mathbf{x}(t)) \tag{2.49}$$

This adaptive state transition mechanism represents a significant advancement over traditional approaches, allowing the model to dynamically adjust its temporal processing based on input content. Rather than processing all temporal information uniformly, Mamba can selectively update its state based on the relevance and importance of incoming information, making it particularly effective for handling the varying levels of motion activity present in natural videos.

For practical implementation, the continuous system is discretized using zero-order hold (ZOH):

$$\begin{aligned}\bar{\mathbf{A}}_t &= e^{\Delta t \mathbf{A}_t} \\ \bar{\mathbf{B}}_t &= (\Delta t \mathbf{A}_t)^{-1} (e^{\Delta t \mathbf{A}_t} - \mathbf{I}) \Delta t \mathbf{B}_t\end{aligned}\tag{2.50}$$

where Δt represents a learnable timestep parameter that enables the model to adapt its temporal resolution to the specific requirements of different video understanding tasks.

Bidirectional Processing and Selective State Updates

Mamba’s architecture incorporates powerful bidirectional processing capabilities, enabling it to simultaneously analyze information flowing in both temporal directions. This is formalized through parallel forward and backward state space models:

$$\begin{aligned}\mathbf{Y}_{\text{fwd}} &= \text{SSM}(\mathbf{X}_{1:T}) \\ \mathbf{Y}_{\text{bwd}} &= \text{SSM}(\mathbf{X}_{T:1}) \\ \mathbf{Y} &= \text{Linear}(\mathbf{Y}_{\text{fwd}} \oplus \mathbf{Y}_{\text{bwd}})\end{aligned}\tag{2.51}$$

This bidirectional processing is particularly valuable in action quality assessment and complex motion analysis, where understanding the complete context of a movement often requires both forward and backward temporal information. The architecture’s ability to maintain long-term dependencies while efficiently processing temporal information makes it especially effective for applications requiring fine-grained analysis of extended sequences.

The selective state update mechanism achieves remarkable computational efficiency, demonstrating linear complexity $\mathcal{O}(T)$ compared to the quadratic complexity $\mathcal{O}(T^2)$ of transformer-based approaches:

$$\text{FLOPs}_{\text{Mamba}} = T(4N^2 + 3N) \quad \text{vs} \quad \text{FLOPs}_{\text{Transformer}} = 4T^2D + 2TD^2\tag{2.52}$$

This efficiency is not merely a theoretical advantage but translates directly to practical benefits in video processing applications, enabling the analysis of longer sequences and higher frame rates without requiring prohibitive computational resources.

Spatiotemporal Adaptations

Early applications [53] of Mamba to video understanding tasks have demonstrated promising results across multiple domains. The architecture implements sophisticated spatiotemporal processing through factorized computation:

$$\mathbf{F}_{\text{out}} = \text{SSM}_{\text{time}}(\text{SSM}_{\text{space}}(\mathbf{F}_{\text{in}})) \quad (2.53)$$

This factorized processing, combined with hierarchical state compression:

$$\mathbf{H}_l = \text{SSM}(\text{Downsample}(\mathbf{H}_{l-1})) \quad (2.54)$$

enables efficient handling of both fine-grained motion details and long-term temporal dependencies. The architecture’s success across various video understanding tasks, including action quality assessment, motion analysis, and behavioral understanding, positions it as a promising direction for advancing the state-of-the-art in video understanding tasks.

2.3.6 Future Directions and Hybrid Architectures

The architectural evolution of video understanding is converging toward hybrid paradigms that synergistically combine complementary modeling philosophies. Contemporary research has identified the potential of integrating selective state dynamics from Mamba’s SSM framework, global attention mechanisms from transformer architectures, and local feature induction from convolutional networks. This synthesis represents a fundamental advance in addressing the core tension in video understanding – the challenge of balancing long-range dependency modeling with computational efficiency while preserving spatial-semantic fidelity.

The integration of these architectural components can be formalized through a gated feature composition mechanism:

$$\mathbf{F}_{\text{hybrid}} = \alpha \cdot \text{SSM}(\mathbf{X}) + \beta \cdot \text{Attention}(\mathbf{X}) + \gamma \cdot \text{Conv3D}(\mathbf{X}) \quad (2.55)$$

where α , β , and γ represent learnable mixing coefficients that are dynamically conditioned on input characteristics. This adaptive fusion enables the model to leverage the strengths of each architectural component based on the specific requirements of different video segments. Early implementations of this approach, exemplified by *MambaFormer* [76], have demonstrated promising results, achieving significant accuracy improvements over pure SSM architectures on benchmark datasets like Kinetics-600 while maintaining much of Mamba’s computational efficiency advantages.

2.4 Action Recognition: Foundations and Evolution

Challenge 1: Motion Temporal Variability in Action Recognition. This chapter addresses the fundamental challenge of motion temporal variability - the phenomenon where identical actions exhibit dramatically different temporal patterns across different instances, contexts, or performers. Unlike image classification, which focuses on static visual features, action recognition requires modeling complex temporal dynamics across variable time scales. A professional tennis player’s serve involves explosive, precise movements completed in under a second, while an amateur’s serve might contain hesitations, repositioning, and slower execution of the same fundamental action. Current methodologies often develop spurious correlations with static contextual elements (background objects, scene setting) rather than capturing genuine motion patterns that define actions. This static bias problem leads to poor generalization when background contexts change, limiting real-world effectiveness. Traditional fixed-window approaches struggle to differentiate between motion variations that change the action’s identity and stylistic differences in performing the same action.

Fundamental Concepts in Video-based Action Recognition. To understand the technical approaches discussed in this chapter, several key concepts require clarification. **Temporal modeling** refers to how computational methods capture the sequential nature of actions - understanding not just what appears in individual frames, but how visual patterns evolve over time. **Spatiotemporal features** combine spatial information (what objects and patterns appear in each frame) with temporal information (how these patterns change across frames). **Optical flow** represents the apparent motion of pixels between consecutive frames, serving as a fundamental representation of movement in video analysis. **3D convolutions** extend traditional 2D image convolutions to include the time dimension, allowing neural networks to process video volumes directly rather than treating videos as sequences of independent images. These concepts form the foundation for understanding how computational systems can learn to recognize human actions from video data.

Action recognition represents a fundamental challenge in video understanding, requiring the analysis of both spatial content within individual frames and temporal dynamics across frame sequences. Given a video sequence $\mathbf{V} \in \mathbb{R}^{T \times H \times W \times 3}$, where T denotes temporal length and $H \times W$ represents spatial dimensions, the goal is to learn a mapping function:

$$f : \mathbb{R}^{T \times H \times W \times 3} \rightarrow \mathcal{Y} \quad (2.56)$$

where $\mathcal{Y}$ represents the space of possible action categories. This mapping must effectively capture both spatial semantics and temporal evolution:

$$P(y|\mathbf{V}) = f_{\text{temporal}}(f_{\text{spatial}}(\{\mathbf{I}_t\}_{t=1}^T)) \quad (2.57)$$

where $\{\mathbf{I}_t\}_{t=1}^T$ represents the sequence of video frames.

2.4.1 Traditional Methods: The Era of Hand-crafted Features

Understanding Hand-crafted Feature Extraction. Before deep learning, researchers designed specific algorithms to extract meaningful features from videos. **Hand-crafted features** are manually designed representations based on domain expertise, contrasting with learned features that neural networks discover automatically. **Dense trajectories** track the movement of numerous points across video frames, creating paths that represent motion patterns. **Optical flow estimation** calculates the apparent motion of pixels between consecutive frames using mathematical optimization - essentially determining how each pixel has moved from one frame to the next. **Fisher Vector encoding** transforms variable-length sequences of features into fixed-size representations by modeling how features deviate from a learned statistical model, enabling standard machine learning algorithms to process video data of different lengths.

Prior to the deep learning revolution, traditional approaches dominated action recognition, with the Dense Trajectories (DT) and improved Dense Trajectories (iDT) methods proposed by Wang et al. [97, 98] representing the pinnacle of hand-crafted feature approaches. As illustrated in Figure 2.7, these methods operate through a systematic pipeline of motion analysis and feature extraction.

The core innovation of iDT lies in its sophisticated motion trajectory analysis. Given consecutive frames $\mathbf{I}_t$ and $\mathbf{I}_{t+1}$, the method first estimates dense optical flow:

$$\mathbf{w}(x, y, t) =_{\mathbf{w}} \sum_{x, y} \|\mathbf{I}_{t+1}(x + u, y + v) - \mathbf{I}_t(x, y)\|_2^2 + \alpha \|\nabla \mathbf{w}\|_2^2 \quad (2.58)$$

where $\mathbf{w}(x, y, t) = (u, v)$ represents the optical flow vector at position (x, y) and time t . The method then compensates for camera motion through homography estimation:

$$\mathbf{H}_{t,t+1} = \mathbf{H} \sum_{p \in \mathcal{P}} \|\mathbf{p}_{t+1} - \mathbf{H}\mathbf{p}_t\|_2^2 \quad (2.59)$$

where $\mathcal{P}$ represents the set of matched SURF keypoints between frames.

Along these motion-compensated trajectories, iDT extracts a rich set of descriptors:

$$\mathbf{f}_{\text{trajectory}} = \{\Delta \mathbf{P}_t\}_{t=1}^L, \quad \Delta \mathbf{P}_t = (\mathbf{P}_{t+1} - \mathbf{P}_t) / (\|\mathbf{P}_{t+1} - \mathbf{P}_t\|_2) \quad (2.60)$$

These trajectory features are complemented by local descriptors:

- Histogram of Oriented Gradients (HOG): $\mathbf{f}_{\text{HOG}} = \text{hist}(\nabla \mathbf{I})$
- Histogram of Optical Flow (HOF): $\mathbf{f}_{\text{HOF}} = \text{hist}(\mathbf{w})$
- Motion Boundary Histogram (MBH): $\mathbf{f}_{\text{MBH}} = \text{hist}(\nabla \mathbf{w})$

The final representation is obtained through Fisher Vector encoding:

$$\Phi(\mathbf{X}) = \begin{bmatrix} \sum_{t=1}^T \gamma_t(k) \frac{\partial \log u_k}{\partial \mu_k} \\ \sum_{t=1}^T \gamma_t(k) \frac{\partial \log u_k}{\partial \sigma_k} \end{bmatrix} \quad (2.61)$$

While these traditional methods achieved remarkable success for their time, they faced significant limitations in both computational efficiency and feature dimensionality. The high-dimensional Fisher Vector representations often exceeded the size of the original videos, and the sequential nature of the processing pipeline made real-time analysis impractical. Despite these limitations, the insights gained from these approaches, particularly in motion analysis and trajectory-based features, continue to influence modern deep learning architectures for action recognition.

2.4.2 Deep Neural Architectures in Video Understanding

The Transition from Hand-crafted to Learned Features. The emergence of deep learning marked a fundamental shift from manually designing features to automatically learning optimal representations from data. Unlike hand-crafted approaches that relied

Dense Trajectories (DT) and improved Dense Trajectories (iDT)

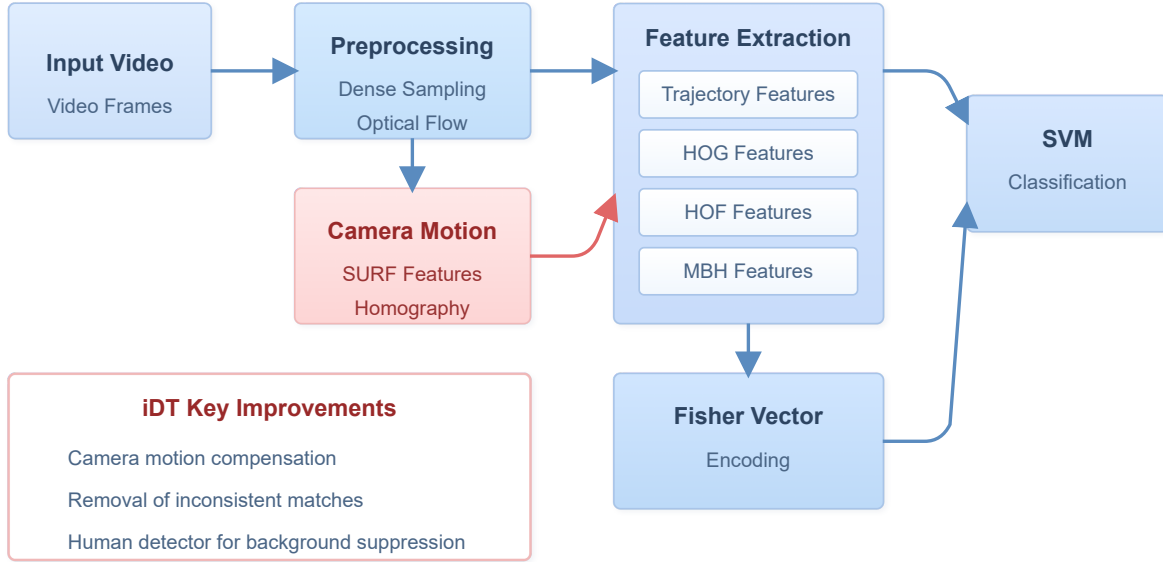


Figure 2.7: Improved Dense Trajectories (iDT) method for extracting motion features from videos. The approach tracks dense points through optical flow, extracts trajectory shape, HOG, HOF, and MBH descriptors along these trajectories, and compensates for camera motion to improve feature quality. This hand-crafted feature extraction technique was a breakthrough in pre-deep learning video understanding. The figure is from [15].

on domain expertise to define relevant features, **deep neural networks** automatically discover hierarchical patterns through multiple layers of processing. Each layer learns increasingly complex representations, from simple edges and motion patterns in early layers to complex action concepts in deeper layers. This automatic feature learning capability eliminates the need for manual feature engineering while often achieving superior performance through end-to-end optimization.

The quest for effective spatiotemporal feature learning in video understanding presents unique challenges that extend beyond traditional image analysis. Video data inherently involves the complex interplay of spatial and temporal dimensions, requiring architectures capable of processing high-dimensional, dynamic information across multiple scales. While deep learning has revolutionized image processing since the early 2010s, its application to video analysis has demanded significant architectural innovations to address the distinct

challenges of temporal reasoning in sequential visual data.

The evolution of deep architectures for video understanding has progressed through three distinct phases, each marked by fundamental breakthroughs in spatiotemporal modeling. This progression reflects the field’s continuous effort to balance computational efficiency with representational capacity while addressing the unique challenges of temporal reasoning in sequential visual data.

3D Convolutional Foundations for Spatiotemporal Learning

Understanding 3D Convolutions for Video Processing. Traditional 2D convolutions process single images by applying filters across height and width dimensions. **3D convolutions** extend this concept to video by adding the temporal dimension, creating volumetric filters that simultaneously capture spatial patterns (what objects appear) and temporal patterns (how they move over time). Think of 3D convolutions as sliding a cube-shaped filter through a video volume, where the cube examines multiple frames simultaneously to detect spatiotemporal patterns like "a hand moving upward" or "a ball bouncing." This unified processing approach contrasts with earlier methods that processed spatial and temporal information separately, enabling more effective learning of action-specific motion patterns.

The conceptual breakthrough of extending image-based 2D convolution to volumetric spacetime operations established critical foundations for video understanding. The seminal work by Ji et al. [46] in 2012 first demonstrated the effectiveness of 3D CNNs for human action recognition, marking a significant milestone in direct spatiotemporal feature learning. This was followed by Tran et al.’s [93] comprehensive study in 2015, which achieved notable performance improvements on standard benchmarks: 82.3% accuracy on UCF101 [89] and 56.8% on HMDB51 [51] dataset.

Building upon these foundations, I3D (Inflated 3D ConvNets) [12] pioneered the inflation of pretrained ImageNet kernels into third-order tensors, enabling joint learning of spatial features and temporal dynamics through true 3D convolutions. As shown in Figure 2.4, this architectural innovation demonstrated the importance of explicitly modeling appearance and motion cues through its dual-stream implementation—processing both RGB frames and computed optical flow. The effectiveness of this approach was evidenced by significant performance improvements: 95.6% on UCF101, 74.8% on HMDB51, and 71.1% on Kinetics400 [49] dataset.

Subsequent innovations continued to refine the 3D convolution paradigm. R(2+1)D [95] introduced temporal factorization through decoupled spatial and temporal convolutions,

while SlowFast [28] advanced hierarchical feature learning through complementary pathways operating at different temporal resolutions. ResNet3D achieved 94.5% accuracy on UCF101 in 2018, and the SlowFast network reached 79.8% on Kinetics400 in 2019, demonstrating the continued evolution and effectiveness of 3D convolutional architectures.

These architectures established critical design principles for spacetime feature hierarchy construction, though they faced inherent challenges in computational complexity and temporal context modeling across extended sequences. The progression from basic 3D convolutions to sophisticated multi-stream architectures reflects the field’s growing understanding of the unique requirements for effective video understanding.

Efficient Temporal Modeling through 2D Architectural Adaptation

Balancing Efficiency and Temporal Understanding. While 3D convolutions effectively model spatiotemporal patterns, they require significantly more computation than 2D convolutions. This computational burden motivated researchers to develop efficient alternatives that enhance standard 2D image processing networks with temporal reasoning capabilities. **Two-stream architectures** process spatial appearance (RGB frames) and motion information (optical flow) through separate networks, then combine their predictions. **Temporal segment networks** sample representative frames from different video segments rather than processing all frames, dramatically reducing computation while maintaining temporal coverage. **Temporal shift modules** achieve temporal modeling with zero additional parameters by strategically shifting feature channels between adjacent time steps, creating implicit temporal connections within standard 2D networks.

The computational demands of full 3D convolution operations motivated a parallel research trajectory focused on enhancing standard 2D CNNs with efficient temporal reasoning mechanisms. This approach sought to maintain the computational benefits of 2D architectures while incorporating sophisticated temporal modeling capabilities. The evolution of these enhanced 2D paradigms marks a significant milestone in balancing efficiency with temporal representation power.

A foundational breakthrough came with Simonyan and Zisserman’s Two-Stream Networks [87], which decomposed video understanding into parallel spatial and temporal pathways:

$$\mathbf{y} = \text{softmax}(\mathbf{W}_s \mathbf{f}_s + \mathbf{W}_t \mathbf{f}_t) \quad (2.62)$$

where $\mathbf{f}_s$ and $\mathbf{f}_t$ represent spatial (RGB) and temporal (optical flow) features respectively. This architecture achieved significant improvements over hand-crafted features like IDT, reaching 88.0% accuracy on UCF101 and 59.4% on HMDB51.

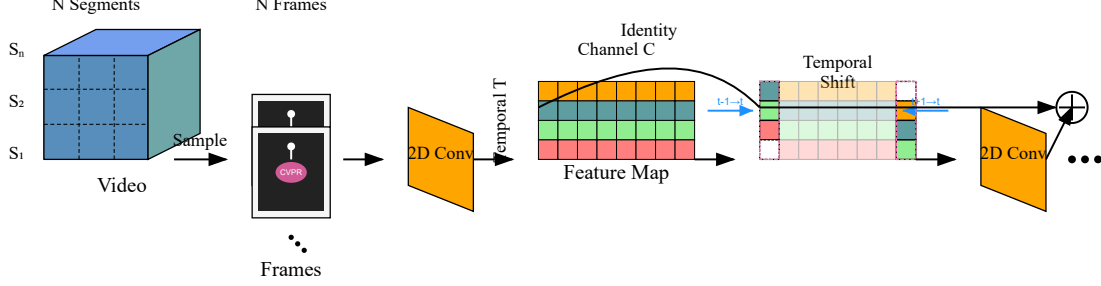


Figure 2.8: The Temporal Shift Module architecture (adapted from [57]). The module implements temporal modeling through channel shifting, where a portion of channels is shifted forward or backward along the temporal dimension. This zero-cost operation effectively creates temporal receptive fields without introducing additional parameters, enabling efficient temporal modeling while maintaining the computational benefits of 2D CNNs.

Building on this foundation, the Temporal Segment Network (TSN) [101] introduced a sparse temporal sampling strategy that effectively captured long-range dependencies while maintaining computational efficiency:

$$\mathcal{F}(\mathbf{V}) = \mathcal{G}(\mathcal{F}_1(\mathbf{T}_1), \mathcal{F}_2(\mathbf{T}_2), \dots, \mathcal{F}_K(\mathbf{T}_K)) \quad (2.63)$$

where $\mathbf{T}_k$ represents strategically sampled temporal segments and $\mathcal{G}$ denotes a consensus function. This segmented processing approach proved particularly effective for long-duration videos, achieving 94.0% on UCF101, 68.5% on HMDB51, and 73.9% on Kinetics400.

A significant architectural innovation emerged with the Temporal Shift Module (TSM) [57], which introduced a parameter-free channel shifting mechanism. For an input tensor $\mathbf{X} \in \mathbb{R}^{N \times C \times T \times H \times W}$ representing batch size (N), channels (C), temporal frames (T), and spatial dimensions (H, W), TSM implements temporal modeling through strategic feature displacement:

$$\mathbf{X}'_{t,c} = \begin{cases} \mathbf{X}_{t-1,c} & \text{if } c \in \mathcal{C}_{\text{forward}} \\ \mathbf{X}_{t+1,c} & \text{if } c \in \mathcal{C}_{\text{backward}} \\ \mathbf{X}_{t,c} & \text{otherwise} \end{cases} \quad (2.64)$$

where $\mathcal{C}_{\text{forward}}$ and $\mathcal{C}_{\text{backward}}$ represent channel sets for forward and backward shifting respectively. As illustrated in Figure 2.8, this elegant design creates implicit temporal receptive fields without additional parameters.

The evolution of 2D architectural adaptations also saw significant contributions from other approaches. TDD (Trajectory-pooled Deep-convolutional Descriptors) [99] achieved

90.3% on UCF101 and 63.2% on HMDB51 through trajectory-guided feature pooling, while LRCN (Long-term Recurrent Convolutional Networks) [21] reached 82.3% on UCF101 by combining CNN features with recurrent temporal modeling. Late fusion strategies, as demonstrated in [29], further improved performance through sophisticated feature integration (92.5% UCF101, 65.4% HMDB51).

These architectural innovations demonstrated that carefully designed 2D adaptations could approximate the temporal modeling capabilities of 3D convolutions while maintaining significant computational advantages. The success of these approaches established important design principles for efficient video understanding, particularly the value of sparse temporal sampling and feature shifting strategies in balancing computational cost with temporal modeling capacity.

Transformer-Based Spacetime Attention Paradigms

Understanding Attention Mechanisms in Video Processing. **Attention mechanisms** allow models to dynamically focus on relevant parts of the input while processing. In video understanding, attention helps determine which frames, spatial regions, and temporal moments are most important for recognizing an action. Unlike convolutions that process local neighborhoods uniformly, **self-attention** can relate any part of the video to any other part, enabling long-range temporal dependencies. For example, when recognizing a tennis serve, attention might focus on the ball toss at the beginning and the racket contact later, even if these events are separated by many frames. **Divided space-time attention** processes spatial and temporal dimensions separately—first determining which spatial regions are important within each frame, then determining which frames are important for the overall action—providing computational efficiency while maintaining modeling power.

The introduction of transformer architectures to video understanding marked a fundamental paradigm shift in the field. Originally designed for natural language processing [96], transformers demonstrated remarkable capabilities in modeling long-range dependencies through self-attention mechanisms. The successful adaptation of transformers to computer vision, particularly through the Vision Transformer (ViT) [22], laid the groundwork for their extension to video understanding tasks, establishing that attention-based architectures could effectively process visual data when provided with sufficient training examples.

The pioneering Vision Transformer architecture revolutionized visual processing by decomposing images into sequences of patches:

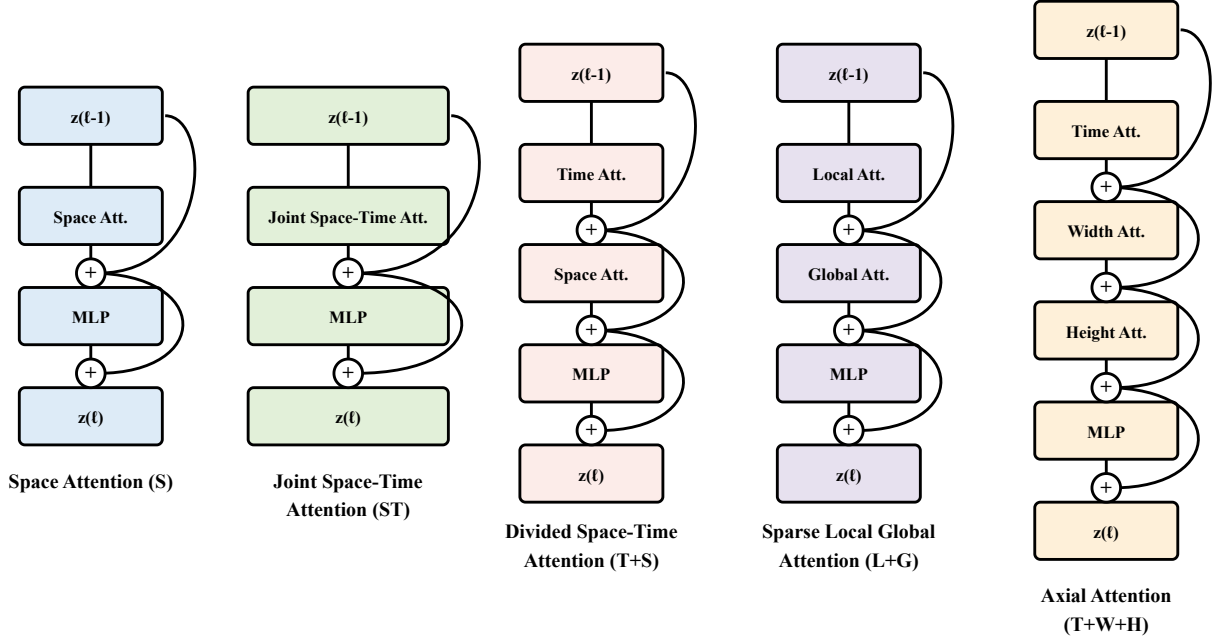


Figure 2.9: Different variants of space-time attention mechanisms in TimeSformer (adapted from [6]). The figure illustrates various attention strategies: (a) space-only attention, (b) joint space-time attention, (c) divided space-time attention, and (d) sparse local attention, showcasing the evolution of attention mechanisms for efficient video processing.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1 \mathbf{E}; \mathbf{x}_p^2 \mathbf{E}; \dots; \mathbf{x}_p^N \mathbf{E}] + \mathbf{E}_{\text{pos}} \quad (2.65)$$

where each patch is linearly projected and combined with positional embeddings. This patch-based approach naturally extended to video data, where the additional temporal dimension introduced both new challenges and opportunities for architectural innovation.

Video Vision Transformers (ViViT) [3] emerged as one of the first successful adaptations of transformers to video understanding. ViViT introduced a factorized self-attention mechanism that efficiently processed both spatial and temporal dimensions by decomposing video clips into tubelets - sequences of image patches spanning multiple frames. This factorized approach enabled the model to capture complex spatiotemporal dependencies while managing computational complexity through separate spatial and temporal processing streams.

TimeSformer [6] further refined this approach through divided space-time attention, as illustrated in Figure 2.9. By decomposing the attention operation into separate spatial and

temporal components:

$$\mathbf{F}_{\text{out}} = \text{Attention}_{\text{temporal}}(\text{Attention}_{\text{spatial}}(\mathbf{F}_{\text{in}})) \quad (2.66)$$

this factorized processing strategy achieved an optimal balance between computational efficiency and modeling capacity. The key insight behind this design, as shown in Figure 2.9(c), was that separating spatial and temporal attention operations could maintain the model’s ability to capture complex temporal dependencies while significantly reducing computational overhead. This approach proved particularly effective for action recognition tasks, demonstrating that dedicated temporal modeling could match or exceed the performance of full spatiotemporal attention computation while requiring substantially fewer computational resources.

The evolution of video transformers continued with architectures like MViT (Multiscale Vision Transformers) [25] and Video Swin Transformer [68], each introducing innovative solutions to the challenges of video processing. MViT established a hierarchical structure that progressively reduced spatial resolution while expanding channel capacity, effectively modeling both fine and coarse-grained features. The Video Swin Transformer adapted the shifted window attention pattern to the video domain, providing a computationally efficient approach to processing high-resolution video inputs while maintaining the ability to model long-range dependencies.

These transformer-based architectures offered several fundamental advantages over conventional 3D CNNs:

First, their attention mechanisms could more effectively capture long-range dependencies across both spatial and temporal dimensions, enabling better modeling of complex action patterns. Second, their hierarchical representations naturally accommodated multi-scale temporal phenomena, from fine-grained motion patterns to long-term action dynamics. Third, their self-attention mechanisms provided interpretable representations of how different video segments contributed to final predictions, offering insights into the model’s decision-making process.

The success of transformer architectures in video understanding has catalyzed continued innovation in architectural design. Current research explores more sophisticated attention mechanisms, efficient computational strategies, and enhanced pretraining approaches. This ongoing evolution suggests that transformer-based architectures will continue to play a central role in advancing the field of video understanding, particularly as computational resources and available training data continue to expand.

2.5 Temporal Action Localization

Challenge 2: Context-Dependency in Temporal Action Localization. This section addresses the fundamental challenge of context-dependency in temporal action localization - the requirement that video segment interpretation depends critically on global temporal context. Current approaches lack sufficient enhancement of global context features, leading to models that do not have a comprehensive understanding of the overall content and are unable to accurately detect action boundaries within video sequences. The interpretation of any video segment fundamentally depends on understanding the global context of the entire video sequence. Without proper global context modeling, the same visual pattern may be misinterpreted as different actions, and models struggle to identify precise temporal boundaries where actions begin and end. A hand moving toward an object could indicate reaching, waving, or gesturing, but accurate interpretation requires understanding the global context of the video sequence. Current attention mechanisms applied to temporal action localization tasks show limited effectiveness in this specialized application scenario, particularly in capturing the global context necessary for accurate boundary detection.

Key Concepts in Temporal Action Localization. Understanding temporal action localization requires familiarity with several critical concepts. **Untrimmed videos** are raw footage containing both action segments and background content, representing realistic video data where actions occur sporadically within longer sequences. **Temporal boundaries** refer to the precise start and end times of actions within continuous video streams - a crucial challenge since even human annotators may disagree on exact boundary locations. **Temporal proposals** are candidate action segments generated by algorithms, similar to object detection bounding boxes but in the time dimension. **Two-stage architectures** separate the localization task into proposal generation (finding candidate segments) and classification (determining action categories), while **end-to-end architectures** jointly optimize both tasks simultaneously. **Boundary matching** refers to techniques that assess the quality of temporal boundaries by evaluating the correspondence between predicted start/end points and ground truth annotations. These concepts form the foundation for understanding how systems can precisely locate actions within continuous video streams.

2.5.1 Task Definition

Temporal Action Localization (TAL) aims to precisely identify the temporal boundaries and categories of specific actions within untrimmed video streams. The technical challenges primarily stem from the spatiotemporal diversity of action instances: the same action

may exhibit significant appearance variations across different scenarios, while boundaries between different actions often display inherent ambiguity.

2.5.2 Technical Evolution

Evolution of Temporal Localization Approaches. The field has evolved from computationally expensive sliding window approaches to sophisticated architectures that efficiently process long video sequences. **Sliding window methods** exhaustively evaluate all possible temporal segments, leading to quadratic computational complexity that becomes prohibitive for long videos. **Post-processing techniques** like Non-Maximum Suppression (NMS) remove redundant overlapping detections by selecting the highest-confidence proposal among overlapping candidates. **Multi-scale feature capture** enables models to detect actions of varying durations by processing features at different temporal resolutions. **Attention mechanisms** in this context help models focus on relevant temporal regions while suppressing irrelevant background content, though their effectiveness in capturing global context for boundary detection remains limited.

While traditional approaches relied on hand-crafted features such as Histogram of Oriented Gradients (HOG) and Histogram of Optical Flow (HOF), the advent of deep learning has catalyzed a paradigm shift in this domain. Current TAL methodologies can be categorized into three main technical paradigms, as illustrated in Figure 2.10:

Segment-Based Classification Methods

These early approaches adopt a sliding window strategy, generating fixed-length video segments (typically 16 frames) and classifying each segment using temporal convolutional networks (TCN) or LSTM architectures. Representative works include S-CNN [59] and SSN [123], which employ post-processing techniques such as non-maximum suppression and probability smoothing. However, their quadratic computational complexity $\mathcal{O}(N^2)$ limits efficiency for long videos.

Two-Stage Architectures

This paradigm decouples boundary detection and classification tasks, with the first stage generating temporal proposals and the second stage performing fine-grained classification. Notable examples include BSN [60], which introduces boundary-sensitive networks, and

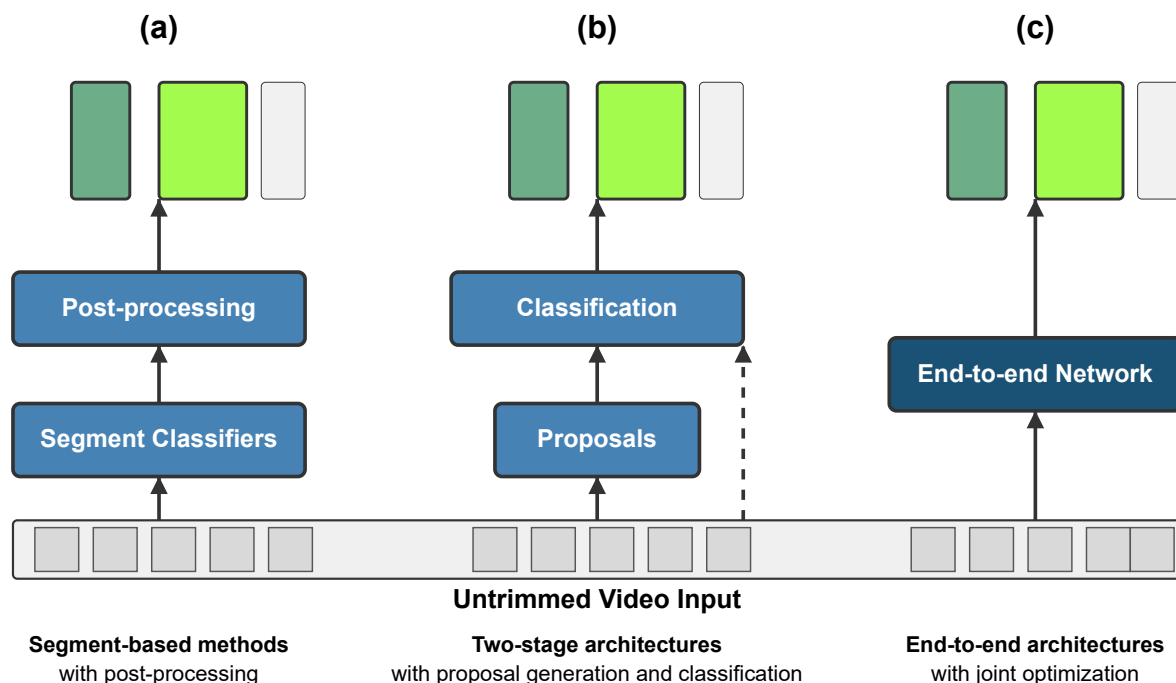


Figure 2.10: Classification of temporal action localization approaches (adapted from [7]): (a) Segment-based methods with post-processing, which process fixed-length segments independently, (b) Two-stage architectures with separate proposal generation and classification stages, (c) End-to-end architectures that jointly optimize proposal generation and classification.

BMN [58], which enhances proposal quality through boundary matching mechanisms. Their modular design significantly reduces memory requirements.

End-to-End Architectures

The most recent approaches jointly optimize proposal generation and classification. Examples include PBRNet [62] with pyramid structures for multi-scale feature capture and GTAN [70] leveraging attention mechanisms. Despite theoretical advantages, these methods face practical limitations due to high memory requirements, often requiring up to 12GB GPU memory for processing 512-frame videos with TSN-ResNet50 backbone.

2.5.3 Boundary Matching Networks: A Two-Stage Approach

Understanding Boundary Matching and Confidence Maps. The Boundary Matching Network introduces sophisticated techniques for temporal proposal generation. **Confidence maps** are two-dimensional representations where each element indicates the likelihood that an action occurs at a specific start time with a particular duration - essentially creating a dense prediction space for all possible temporal segments. **Boundary matching mechanisms** evaluate how well predicted start and end boundaries align with actual action boundaries by combining temporal context with boundary probabilities. **Duration priors** leverage dataset statistics to limit the search space to realistic action lengths, improving both efficiency and accuracy. **Multi-task learning frameworks** simultaneously optimize multiple objectives (boundary detection and confidence prediction) to achieve better overall performance than optimizing each task independently.

The Boundary Matching Network (BMN) represents a significant advancement in two-stage temporal action localization methods. As illustrated in Figure 2.11, it introduces an innovative "boundary matching" mechanism that transforms proposal generation into a dense confidence prediction problem. The network architecture consists of three main components that work in concert to achieve efficient temporal action localization.

Given an input video sequence, BMN first employs TSN [101] as the backbone network for frame-level feature extraction, producing a temporal feature sequence $\mathbf{F} \in \mathbb{R}^{T \times C}$, where T denotes the temporal length of the video (number of frames or temporal segments) and C represents the feature dimension extracted by the backbone network. A dual-branch temporal evaluation module then processes these features to predict start and end probabilities independently for each temporal position:

$$\mathbf{P}_s, \mathbf{P}_e = f_{\text{temp}}(\mathbf{F}), \quad \mathbf{P}_s, \mathbf{P}_e \in \mathbb{R}^T \quad (2.67)$$

where $f_{\text{temp}}(\cdot)$ represents the temporal evaluation function that processes the input features $\mathbf{F}$ to generate start probabilities $\mathbf{P}_s$ and end probabilities $\mathbf{P}_e$, with each vector containing T elements corresponding to the likelihood of an action starting or ending at each temporal position.

The core innovation of BMN lies in its boundary-matching confidence map generation. The network constructs a two-dimensional confidence map $\mathbf{M} \in \mathbb{R}^{T \times D_{\max}}$, where each element $\mathbf{M}(s, d)$ represents the confidence score for a temporal action proposal starting at time step s with duration d frames, and $D_{\max}$ is the maximum allowable duration for action proposals. This confidence score is computed through a boundary matching mechanism that combines start/end probabilities with temporal context:

Boundary Matching Network (BMN) Architecture

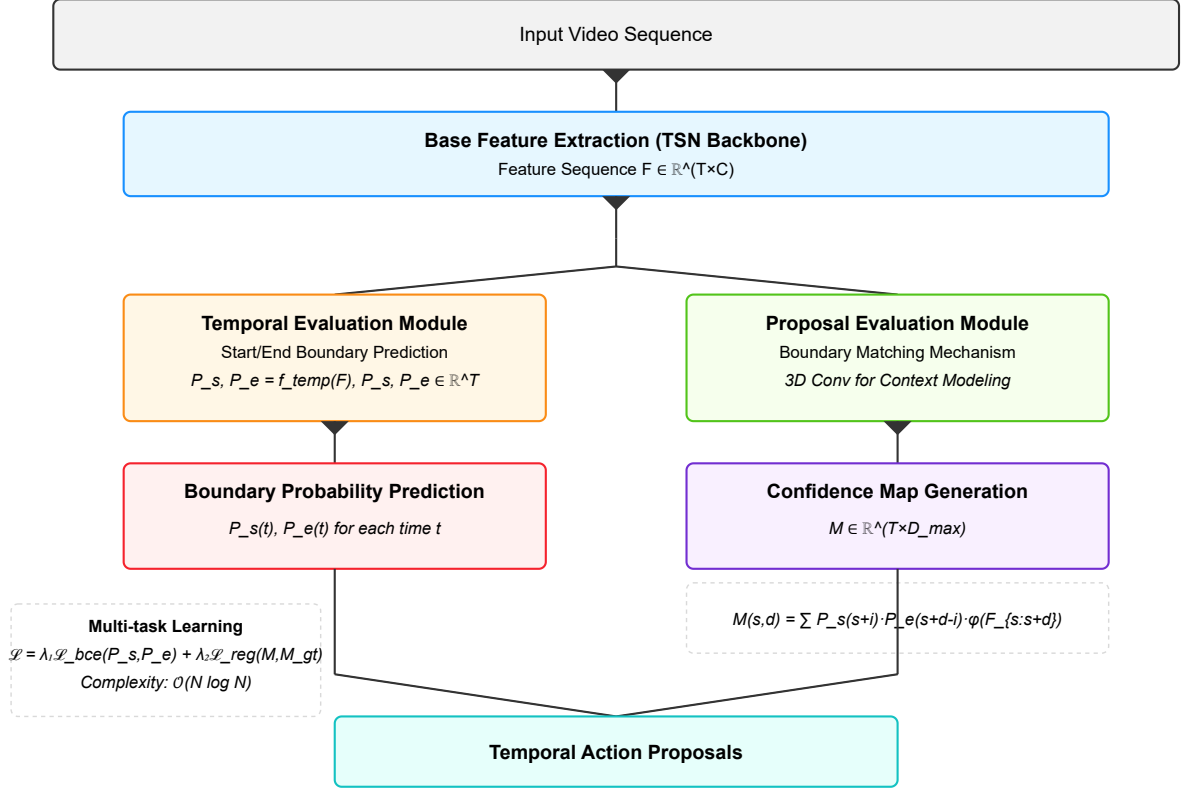


Figure 2.11: The BMN architecture (adapted from [58]). The network consists of three main components: base feature extraction, temporal evaluation module for boundary probability prediction, and proposal evaluation module for confidence map generation.

$$\mathbf{M}(s, d) = \sum_{i=0}^{d-1} \mathbf{P}_s(s+i) \cdot \mathbf{P}_e(s+d-i) \cdot \phi(\mathbf{F}_{s:s+d}) \quad (2.68)$$

where $\mathbf{P}_s(s+i)$ is the start probability at position $s+i$, $\mathbf{P}_e(s+d-i)$ is the end probability at position $s+d-i$, $\mathbf{F}_{s:s+d}$ represents the feature sequence from time s to $s+d$, and $\phi(\cdot)$ represents a temporal evaluation module using 3D convolutions for local context modeling that assesses the actionness of the temporal segment.

BMN introduces several key technical innovations to enhance its efficiency and effectiveness. The architecture employs a novel space-time trade-off strategy by pre-computing matching confidence scores in the confidence map, reducing the time complexity from $\mathcal{O}(N^2)$ for exhaustive proposal enumeration to $\mathcal{O}(N \log N)$ for efficient proposal generation:

$$\text{Complexity}_{\text{BMN}} = \underbrace{T \cdot D_{\max}}_{\text{confidence map generation}} + \underbrace{K \log K}_{\text{Non-Maximum Suppression (NMS)}} \quad (2.69)$$

where the first term $T \cdot D_{\max}$ represents the computational cost of generating the confidence map with T possible start positions and $D_{\max}$ possible durations, and the second term $K \log K$ represents the cost of Non-Maximum Suppression (NMS), where K is the number of generated proposals and NMS is a post-processing technique that removes redundant overlapping proposals by selecting the highest-confidence proposal among overlapping candidates, ensuring diverse and non-redundant final detections.

Additionally, the network employs a duration prior design where $D_{\max}$ is set based on dataset statistics (e.g., 128 frames for THUMOS-14 [42]), effectively balancing computational efficiency with action duration coverage by limiting the search space to realistic action durations observed in the training data.

The training process adopts a multi-task learning framework that combines boundary prediction and confidence regression through a weighted loss function:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{bce}}(\mathbf{P}_s, \mathbf{P}_e) + \lambda_2 \mathcal{L}_{\text{reg}}(\mathbf{M}, \mathbf{M}_{\text{gt}}) \quad (2.70)$$

where λ_1 and λ_2 are weighting parameters that balance the two loss components, $\mathcal{L}_{\text{bce}}(\mathbf{P}_s, \mathbf{P}_e)$ represents the binary cross-entropy loss applied to the start and end probability predictions to ensure accurate boundary detection, $\mathcal{L}_{\text{reg}}(\mathbf{M}, \mathbf{M}_{\text{gt}})$ denotes the regression loss (typically IoU-based) between the predicted confidence map $\mathbf{M}$ and ground truth confidence map $\mathbf{M}_{\text{gt}}$, enabling the network to simultaneously optimize both boundary accuracy and proposal quality assessment.

2.6 Action Quality Assessment: Foundations and Evolution

Challenge 3: Multi-Modal Integration in Action Quality Assessment. This section addresses the third fundamental challenge of multi-modal integration - combining

different types of information streams (visual, audio, temporal, contextual) to achieve comprehensive understanding. This challenge is particularly critical in action quality assessment, which evaluates how well actions are performed rather than simply identifying their occurrence. Unlike action recognition, which typically involves discrete categorical decisions, quality assessment requires nuanced evaluation along continuous scales that reflect expert human judgment. Most current approaches process modalities in isolation before late fusion (combining different modalities only at the final decision stage), missing the complex interrelationships between different signals that expert human evaluators naturally integrate. Late fusion approaches extract features from each modality independently, then combine them only at the final decision stage. This strategy fails to capture temporal synchronization between visual and audio cues, complementary information across modalities, and contradictory signals that require resolution through cross-modal reasoning.

Key Concepts in Action Quality Assessment. Understanding action quality assessment requires familiarity with several specialized concepts. **Action quality assessment (AQA)** evaluates how well actions are performed rather than simply identifying what actions occur, requiring continuous scoring rather than categorical classification. **Multi-modal fusion** refers to combining information from different sources (visual features, audio signals, temporal dynamics) to achieve more comprehensive understanding than any single modality could provide. **Late fusion** processes each modality independently before combining results at the decision stage, while **early fusion** integrates modalities during feature extraction. **Temporal synchronization** refers to the alignment of information across different modalities at corresponding time points - for example, correlating visual motion patterns with audio cues. **Cross-modal reasoning** involves resolving contradictory or complementary information between modalities to make coherent decisions. **Fine-grained temporal analysis** examines subtle motion variations and timing patterns that distinguish high-quality from low-quality action execution. These concepts are essential for understanding how computational systems can evaluate action performance with expert-level precision.

2.6.1 Introduction and Problem Definition

Understanding Quality Assessment vs. Recognition. The distinction between action recognition and quality assessment represents a fundamental shift in computational objectives. **Categorical classification** in action recognition assigns discrete labels (e.g., "jump," "throw," "swing"), while **continuous scoring** in quality assessment produces numerical ratings reflecting performance level. **Expert human judgment** serves as the

gold standard, where trained evaluators consider multiple factors including form, technique, timing, and execution consistency. **Subjective evaluation criteria** vary across domains - gymnastics emphasizes artistic expression and technical difficulty, while surgical assessment prioritizes precision and safety protocols. **Fine-grained analysis** examines subtle variations in movement patterns, joint angles, and temporal dynamics that distinguish excellent from poor performance, requiring models to capture nuanced differences that may be imperceptible to untrained observers.

Action Quality Assessment (AQA) represents a critical research frontier in computer vision that extends beyond traditional action recognition tasks. While action recognition focuses on categorical classification, AQA aims to quantitatively evaluate the execution quality of human actions from video data. This challenging task requires fine-grained analysis of subtle kinematic variations, such as joint angles in gymnastics or tool trajectories in surgical procedures [78].

The applications of AQA span diverse domains including sports analytics (e.g., evaluating Olympic diving performances), healthcare (monitoring rehabilitation progress), and industrial training (ensuring procedural compliance). The complexity of this task stems from the need to understand not just what action is being performed, but how well it is executed, requiring sophisticated temporal modeling and multi-modal analysis capabilities.

2.6.2 Evolution of Technical Approaches

From Hand-crafted to Deep Learning Approaches. The evolution of AQA methodologies mirrors broader trends in computer vision while addressing unique challenges of quality evaluation. **Hand-crafted features** in early AQA systems relied on domain expertise to design relevant quality indicators, such as trajectory smoothness in sports or symmetry measures in rehabilitation exercises. **Support Vector Regression (SVR)** and similar traditional methods provided interpretable models but struggled with the complexity of multi-dimensional quality assessment. **Deep learning architectures** automatically discover hierarchical quality-relevant features, eliminating manual feature engineering while achieving superior performance. **3D convolutions for temporal modeling** capture motion patterns across time, essential for evaluating dynamic actions where timing and sequencing affect quality scores. **Attention mechanisms** in quality assessment help models focus on critical moments and body parts that most influence overall performance evaluation.

The development of AQA methodologies has witnessed significant evolution, particularly with the advent of deep learning techniques. Early approaches relied heavily on

hand-crafted features coupled with traditional regression models such as Support Vector Regression (SVR) [78]. These methods, while interpretable, struggled with generalization due to their limited representational capacity.

The introduction of deep learning architectures marked a paradigm shift in AQA approaches:

$$f_{\text{quality}} = \mathcal{F}_{\theta}(\mathbf{X}), \quad \mathbf{X} \in \mathbb{R}^{T \times H \times W \times C} \quad (2.71)$$

where $\mathbf{X}$ represents the input video sequence and f_{quality} denotes the predicted quality score. Modern approaches have evolved to incorporate sophisticated temporal modeling through various mechanisms:

- **Temporal Modeling:** Advanced architectures like C3D-SVR [117] and TSA [113] improve temporal feature extraction through 3D convolutions and attention mechanisms.
- **Multimodal Integration:** Recent frameworks integrate multiple modalities including RGB, optical flow, and skeleton data to enhance robustness:

$$f_{\text{fusion}} = \phi(f_{\text{rgb}}, f_{\text{flow}}, f_{\text{skeleton}}) \quad (2.72)$$

2.6.3 Challenges and Future Directions

Current Limitations and Research Opportunities. Several fundamental challenges continue to limit the effectiveness of current AQA approaches. **Modality integration** remains problematic when different information sources provide conflicting quality indicators - for example, when visual form appears correct but audio cues suggest timing issues. **Temporal granularity** presents a multi-scale challenge where models must simultaneously capture brief technical errors and overall performance consistency across entire action sequences. **Domain transfer** and generalization difficulties arise when models trained on one sport or activity fail to transfer knowledge to related domains, despite shared principles of movement quality. **Evaluation subjectivity** introduces challenges when human expert scores disagree, requiring models to learn not just average consensus but the range of valid expert opinions. **Real-time processing demands** in practical applications require balancing computational efficiency with the detailed analysis necessary for accurate quality assessment.

Despite significant progress, several fundamental challenges remain in AQA research:

- **Modality Integration:** Effectively combining different input modalities while preserving their unique informational content.
- **Temporal Granularity:** Capturing both fine-grained motion details and long-term action dependencies.
- **Generalization:** Developing models that can transfer knowledge across different action domains and viewpoints.

These challenges motivate the development of more sophisticated architectures that can better capture the nuances of action quality assessment. Future research directions include the exploration of adaptive fusion strategies and the integration of domain-specific knowledge into deep learning frameworks.

2.7 Chapter Summary and Conclusion

This chapter has provided a comprehensive review of video understanding techniques across multiple domains, organized around three fundamental challenges that represent critical barriers to achieving human-like video understanding capabilities. We have examined diverse approaches to improve model effectiveness and efficiency, ranging from explicit modeling of temporal dynamics to enhanced feature representations and attention mechanisms.

Our survey of video understanding architectures has traced their evolution from traditional hand-crafted features to sophisticated deep learning models. We have observed that while 3D CNNs effectively capture spatiotemporal patterns, they struggle with long-range dependencies. Recurrent networks and attention mechanisms address this limitation but introduce computational complexities. Transformers have transformed the field with their self-attention capabilities, yet face scalability challenges with longer videos. The emergence of state space models, particularly the Mamba architecture, offers promising directions for efficient sequence modeling that may overcome these limitations.

Through this review, we have identified how the three core challenges manifest across different domains. **Challenge 1: Motion Temporal Variability** is evident in action recognition models that often rely excessively on static scene elements rather than motion dynamics, limiting their ability to discriminate between fine-grained actions that exhibit different temporal patterns across instances. **Challenge 2: Context-Dependency** appears in temporal localization systems that struggle with precise boundary determination, especially for complex or overlapping actions, due to insufficient global context modeling. **Challenge 3: Multi-Modal Integration** is particularly pronounced in quality

assessment, which remains underexplored, with few approaches effectively modeling the subtle variations that differentiate performance levels through comprehensive integration of different information streams.

These identified challenges directly motivate our integrated research contributions in the subsequent chapters. In Chapter 3, we will address **Challenge 1: Motion Temporal Variability** by introducing a coarse-to-fine-to-finer discriminative learning framework. This approach explicitly models the differences between action and non-action segments to create more robust representations of human activities that focus on genuine motion characteristics rather than static contextual biases.

Chapter 4 will tackle **Challenge 2: Context-Dependency** through a comprehensive action localization system developed for the CVPR ActivityNet Challenge dataset. Building upon the representations introduced in Chapter 3, this system integrates multi-modal features and specialized attention mechanisms to enhance global context understanding for precise action boundary identification within continuous video streams.

Chapter 5 will address **Challenge 3: Multi-Modal Integration** using the Mamba architecture for action quality assessment. This novel framework leverages selective state space models to integrate multiple information streams throughout the assessment process, enabling evaluation of subtle performance variations across diverse domains while capturing the complex interrelationships between different modalities.

Finally, Chapter 6 will synthesize our contributions across all three challenges of temporal understanding, examining how our integrated framework advances the field of video analysis and opens new research directions. We will discuss how our work connects with emerging trends in large language models and video-language models, and reflect on the broader implications of sophisticated video understanding systems as they find applications across diverse domains.

Together, these chapters form our multi-level temporal understanding framework for video analysis, systematically addressing the three fundamental challenges while progressing from recognizing what actions occur to determining when they occur and ultimately to assessing how well they are performed. This integrated approach advances both the theoretical understanding and practical capabilities in AI-driven video analysis.

Chapter 3

Spot the Difference! Temporal Coarse to Fine to Finer Difference Spotting for Action Recognition in Videos

This chapter presents content based on the paper "Spot the Difference! Temporal Coarse to Fine to Finer Difference Spotting for Action Recognition in Videos" [54].

3.1 Introduction

Action recognition in videos represents the foundational first level in our multi-level temporal understanding framework and serves as one of the cornerstone challenges in video understanding. This capability underpins numerous applications spanning surveillance, human-computer interaction, content retrieval, and behavioral analysis. Despite significant progress achieved through large-scale datasets [9, 120, 49, 32, 89] and powerful video representation methods [57, 12, 95, 3, 6], current approaches still struggle with several fundamental challenges that limit their real-world effectiveness.

3.1.1 The Core Challenge: Beyond Simple Classification

Most state-of-the-art action recognition models demonstrate impressive performance on benchmark datasets by learning to map video segments to predefined action categories. However, these models often rely excessively on static scene elements and background context rather than focusing on the dynamic temporal patterns that truly define actions. This phenomenon, known as **shortcut learning**, results in models that perform well in controlled environments but fail to generalize when visual contexts change.

Consider the challenge of distinguishing between “throwing a frisbee” and “meddling with a frisbee”—two activities that may involve identical objects, settings, and even similar body positions, yet differ fundamentally in their temporal execution patterns. Current models frequently struggle with such distinctions because they focus on the presence of objects (the frisbee) and general scene context rather than the specific temporal dynamics that characterize each action.

The existing literature has primarily approached action recognition through direct classification methods, where models are trained to map video segments to predefined action categories. While effective to a degree, this approach neglects the valuable discriminative information available in **untrimmed videos**—raw video sequences that contain both action and non-action segments in their natural temporal order, as opposed to pre-segmented clips that contain only action content.

3.1.2 A New Perspective: Learning Through Discrimination

This chapter introduces a novel paradigm inspired by a simple yet powerful insight: **the most effective way to understand what constitutes an action is to learn how it differs from non-action**. This approach draws inspiration from the cognitive phenomenon observed in “Spot the Difference” games, where our visual system becomes remarkably adept at detecting subtle variations when provided with comparative references.

In the context of video understanding, this translates to a fundamental shift in how we utilize training data. Traditional approaches discard the temporal context surrounding action segments—specifically, the non-action content that often contains preparatory movements, follow-through actions, or contextually related activities. We argue that these discarded segments, precisely because they share visual and semantic continuity with action segments while differing only in the specific activity performed, provide invaluable discriminative information.

3.1.3 Key Terminology and Framework Components

To establish a clear foundation for our approach, we define the following essential concepts that will be used throughout this chapter:

Video Hierarchical Structure: To clarify the data organization in our framework, we establish a clear hierarchical structure:

Complete Video: An entire untrimmed video sequence that contains multiple activities and temporal contexts.

Video Segments: Continuous temporal regions within a complete video, typically spanning several seconds to minutes. Each video is divided into multiple segments based on activity content:

- **Action Video Segments:** Temporal regions containing the target action of interest (e.g., all portions of a video where “throwing frisbee” occurs). These segments can be discontinuous throughout the video and are also known as **trimmed videos** in conventional action recognition datasets, where only these action-containing portions are retained for training.
- **Non-action Video Segments:** All remaining temporal regions that do not contain the target action. These include preparatory movements, follow-through actions, transitions, or contextually related but distinct activities (e.g., “meddling with frisbee” when the target is “throwing frisbee”).

Video Clips: Short temporal windows sampled from video segments, typically containing 16-64 frames (corresponding to approximately 1-10 seconds depending on the frame sampling rate and dataset specifications). Multiple clips are extracted from each segment through sliding window sampling:

- **Action Clips:** Video clips sampled from action video segments, used as positive examples during training.
- **Non-action Clips:** Video clips sampled from non-action video segments, used for discrimination learning.

This hierarchical organization ensures that our discrimination framework operates at the appropriate granularity: learning to distinguish between clips sampled from action segments versus clips sampled from non-action segments, rather than attempting to classify individual frames or entire videos.

Temporal Discrimination: The process of learning to distinguish between action and non-action segments based on their temporal dynamics rather than static visual content. This forms the core principle of our approach.

Coarse-to-Fine-to-Finer Framework: Our hierarchical discrimination approach that operates at three levels of granularity: (1) coarse-level discrimination across different videos, (2) fine-level discrimination within the same video context, and (3) finer-level discrimination within individual segments.

3.1.4 Our Approach and Contributions

Building on this insight, we develop a coarse-to-fine-to-finer discrimination framework that explicitly models the differences between action and non-action segments at multiple levels of granularity. Unlike conventional approaches that focus solely on optimizing classification accuracy on labeled action segments, our method leverages the rich informational contrast between action and non-action content to develop more robust and generalizable representations.

Our framework implements discrimination at three complementary levels of granularity, each addressing a specific aspect of the action recognition challenge:

1. **Coarse-level discrimination** focuses on inter-video differences, employing a novel booster network architecture to enhance the model’s ability to distinguish between action and non-action clips across different videos with diverse visual contexts and backgrounds.
2. **Fine-level discrimination** targets intra-video differences through a strategically designed contrastive learning approach. This level compares video clips sampled from action segments versus non-action segments within the same video, where both clip types share identical visual contexts (same actors, objects, settings) but differ in the specific activities being performed. This forces the model to focus on temporal movement patterns rather than static scene elements.
3. **Finer-level discrimination** addresses the subtlest distinctions through a clip matching mechanism that operates within individual segments. This level detects fine-grained differences between clips sampled from different temporal positions within the same action or non-action segment, enabling the model to capture precise temporal dynamics and movement nuances that characterize authentic action execution.

This multi-level approach progressively refines the model’s understanding of what constitutes an action, forcing it to learn increasingly nuanced discriminative features rather than relying on simplistic contextual cues. Importantly, our framework is **model-agnostic**, designed to enhance the performance of various backbone architectures without adding inference overhead—a critical advantage for real-world deployment.

This approach establishes the foundation of our multi-level temporal understanding paradigm, creating robust action representations that will support more complex temporal understanding tasks in subsequent chapters. The framework effectively addresses the shortcut learning problem by forcing models to focus on temporal dynamics rather than static contextual cues.

3.1.5 Chapter Organization

In the following sections, we first provide a comprehensive problem formulation and present an overview of our method (Section 3.2). We then detail each level of our discrimination framework, explaining the theoretical foundations, implementation details, and integration mechanisms. Section 3.3 introduces our coarse-level discrimination approach for action/non-action classification across videos. Section 3.4 details our fine-level discrimination strategy using contrastive learning within video sequences. Section 3.5 presents our finer-level clip matching techniques for precise temporal localization. Section 3.6 provides comprehensive experimental results and comparative analyses across multiple datasets and model architectures, demonstrating the consistent performance improvements achieved by our approach and providing insights into the mechanisms underlying its effectiveness. Finally, Section 3.7 summarizes our findings and discusses future research directions.

3.2 Temporal Discrimination Learning for Action Recognition

3.2.1 Cognitive Inspiration: The ”Spot the Difference” Paradigm

Do you still remember the childhood puzzle game “Spot the Difference”? This game challenges players to identify subtle variations between two seemingly identical images placed side-by-side. Despite the complexity of the scenes, our visual system is remarkably adept at detecting even minor discrepancies when direct comparison is possible. However, when these images are presented sequentially with time in between, our ability to identify these



Figure 3.1: An example of the “Spot the Difference” game, illustrating how subtle changes can be difficult to detect without direct comparison. In this particular example, the pattern on the girl’s dress changes from a standard design to a four-leaf clover motif. When viewed side-by-side, this difference becomes apparent, but when seen separately, our brain tends to remember only the general presence of a dress with some pattern rather than the specific details of the pattern itself. This cognitive phenomenon forms the foundation of our approach to action recognition—training models to detect differences that would normally be overlooked. The figure is from [44]

differences deteriorates significantly. The challenge intensifies in dynamic versions of this game, where an image transforms gradually over several seconds, making it exceptionally difficult for most observers to pinpoint exactly what has changed.

Interestingly, when provided with a reference image, our capacity to detect these changes—whether it’s an altered pattern on a dress, a repositioned object, or a subtle color variation—dramatically improves. As illustrated in Figure 3.1, such details would be easily overlooked without direct comparison. This phenomenon stems from a fundamental aspect of human visual cognition: our brains are wired to prioritize the primary content and overall meaning of what we see, often at the expense of precise detail recognition.

3.2.2 From Cognitive Insight to Technical Innovation

This cognitive principle directly parallels a significant challenge in video action recognition systems. Current neural network models, when trained using conventional classification loss functions, exhibit a similar tendency to focus predominantly on static scene elements while

neglecting critical temporal variations that distinguish actions from non-actions. Much like our brains generalize visual scenes, these models often capture the broad semantics of a scene while failing to register the subtle temporal dynamics that define specific actions.

The key insight from cognitive science suggests a promising avenue for improving machine learning models: implementing **comparative learning strategies** analogous to the “Spot the Difference” game. In our context, this translates to **temporal discrimination learning**—explicitly training models to compare and contrast action clips and non-action clips that share similar visual contexts but differ in their temporal execution patterns.

Consider the challenge of distinguishing between “throwing” and “pretending to throw” a frisbee. Both activities involve similar body positions, objects, and settings, but differ critically in their temporal execution. By providing models with comparative references (non-action clips), we can enhance their ability to discern these fine temporal details that characterize authentic actions.

3.2.3 Technical Background and Problem Formulation

Action recognition represents a fundamental problem in video understanding that has witnessed remarkable progress due to the development of large-scale datasets [9, 120, 49, 32, 89] and powerful video representation methods [57, 12, 95, 3, 6]. Conventional approaches [28, 57, 82, 12, 27, 104, 116, 69] focus primarily on learning representations from action video segments (trimmed videos) and have demonstrated impressive performance on benchmark datasets.

However, these methods face a significant limitation: real-world videos are generally untrimmed, containing both action and non-action video segments. The annotation process for action recognition datasets involves trimming action segments from longer, continuous videos before categorizing these segments. In standard practice, video clips sampled from non-action segments are systematically discarded, while only clips from action segments are retained with their corresponding labels for training.

What makes these discarded non-action segments particularly valuable is their inherent temporal and contextual relationship with action segments. They frequently maintain strong visual and semantic continuity, featuring the same individuals, objects, settings, and even thematic elements as the action segments. The critical distinction lies solely in the specific activity being performed—action segments showcase the precise action of interest (e.g., “throwing frisbee”), while non-action segments may depict preparatory movements, follow-through actions, or contextually related but distinct activities (e.g., “meddling with frisbee”).

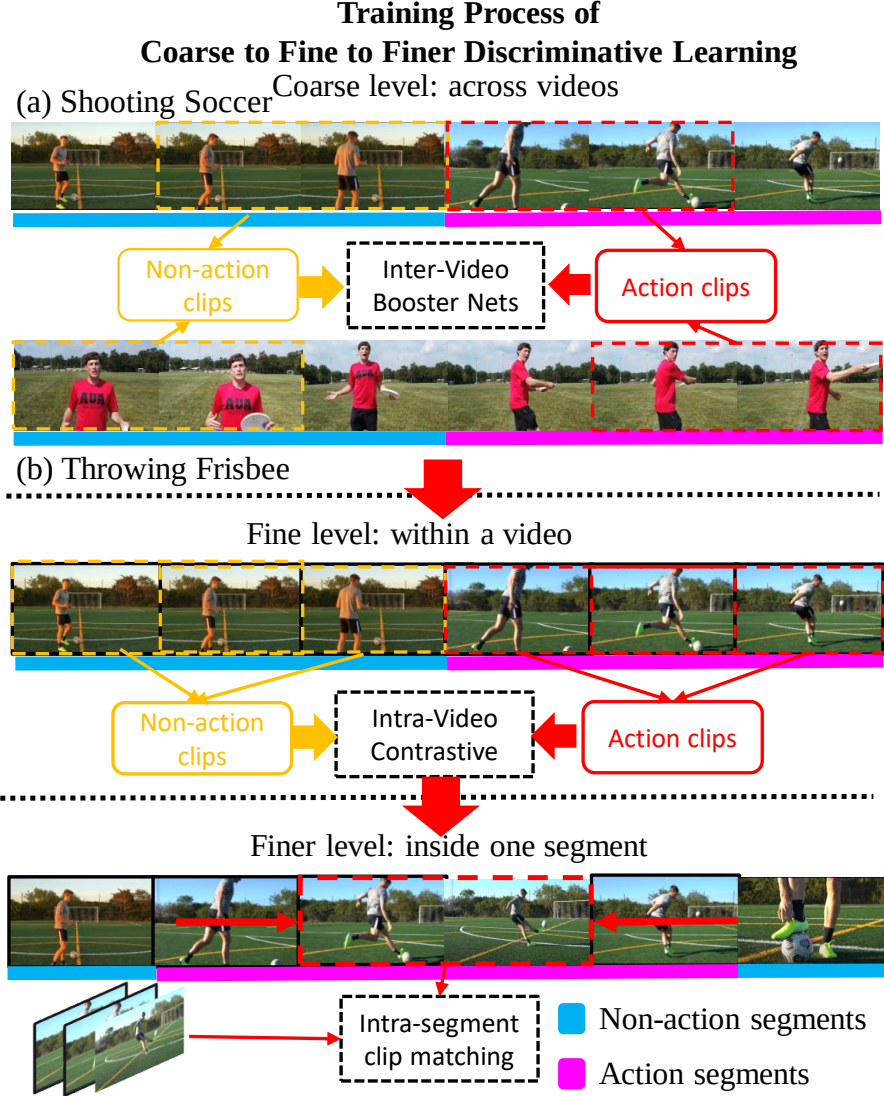


Figure 3.2: Overview of our temporal coarse-to-fine-to-finer action and non-action discriminative learning mechanism. (a) At the coarsest level, our model separates action segments (e.g., "Throwing frisbee," in red) from non-action segments (e.g., "Meddling with frisbee," in blue) across diverse videos. (b) At the fine level, it distinguishes the target action (e.g., "Shooting soccer") from visually similar activities (e.g., "Playing with football") within the same scene. (c) At the finest level, our model precisely localizes action boundaries, identifying when an action starts and ends while ignoring irrelevant motion, shared objects, and static backgrounds.

As illustrated in Figure 3.2, this relationship becomes evident in various scenarios. Non-action segments such as “meddling with a frisbee” temporally precede or follow the actual “throwing frisbee” action. Similarly, “playing with a football” occurs in direct temporal proximity to “shooting soccer” within the same continuous video sequence.

3.2.4 Problem Definition and Method Overview

Action recognition in videos remains challenging due to the complex spatio-temporal nature of human actions, background variability, and subtle inter-class differences. Building on the cognitive insights discussed above, our approach addresses these challenges by learning discriminative features at multiple granularities through temporal discrimination, enabling the model to better distinguish between similar action categories while focusing on movement dynamics rather than static scene elements.

An overview of our proposed method for learning discriminative video representations is shown in Figure 3.3. Our framework implements temporal discrimination learning through a multi-task training mechanism that operates at three distinct yet complementary levels of granularity, each designed to capture different aspects of the discrimination challenge.

1. **Coarse-level discrimination** compares action and non-action clips across different videos with diverse visual contexts using auxiliary booster networks. This allows the model to capture category-level differences and establish broad action/non-action boundaries regardless of background variations.
2. **Fine-level discrimination** employs a novel contrastive learning sampling strategy to incorporate discriminative information between action and non-action clips within the same video sequence. By contrasting clips that share identical visual contexts but differ in temporal activities, this level forces the model to focus on temporally relevant action cues while suppressing irrelevant background information.
3. **Finer-level discrimination** introduces a clip matching mechanism to precisely locate video clips in the temporal dimension and detect subtle temporal variations. This enables the model to focus more on spatio-temporal variations in the moving subject rather than unrelated actions and static scene information.

This hierarchical approach ensures that the model progressively refines its ability to differentiate between action categories at multiple scales, from cross-video comparisons down to fine-grained temporal localization, systematically addressing the shortcut learning problem identified in current action recognition systems.

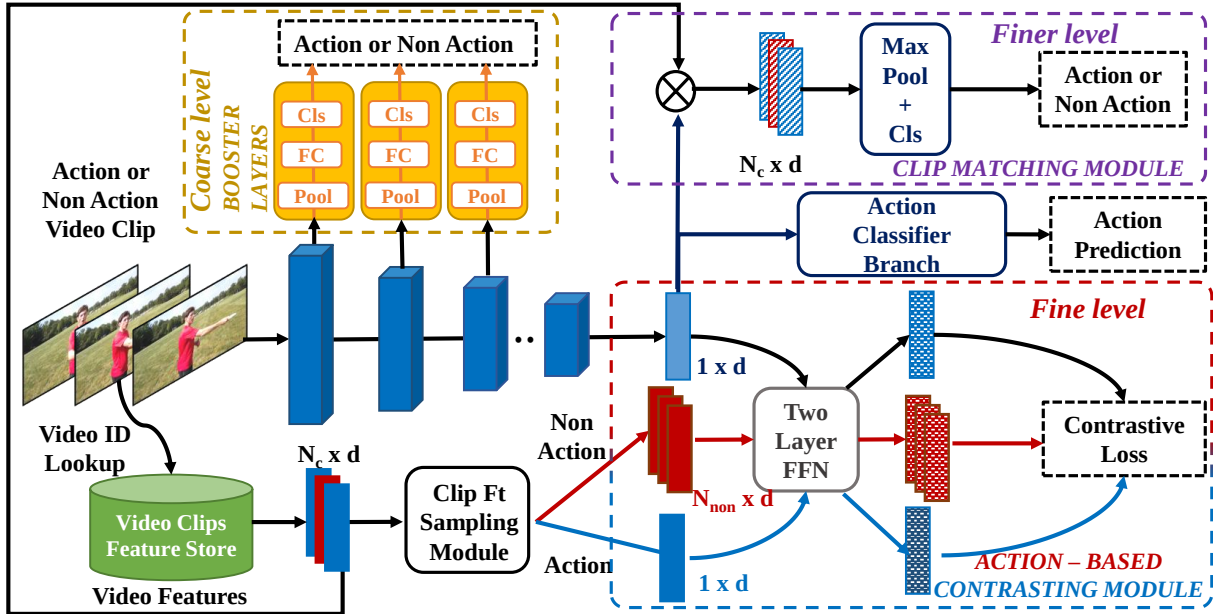


Figure 3.3: Overall schematic of our proposed framework. We employ a base action recognition network on a video clip which may contain action or non-action instances. We introduce booster layers for coarse-level distinction between all action vs non-action clips. For fine-level discrimination, the proposed action-based contrasting module increases the variations of the action and non-action instances from the same video. Finally for the finer level of discrimination, we propose a clip matching module that identifies clips in the same video that are similar to the input clip, and utilizes them to discriminate between action and non-action instances, thus enlarging the differences within a clip.

3.2.5 Model-Agnostic Integration and Advantages

A key advantage of our proposed method is its **model-agnostic** nature—it can seamlessly integrate with existing action recognition frameworks without requiring fundamental architectural redesigns. The framework operates through auxiliary loss functions that complement the standard classification objective, ensuring compatibility with diverse backbone architectures including CNN-based models (TSM-ResNet [57], TANet [69]), attention-based transformers (Timesformer [6]), and hierarchical designs (VideoSwin [68]).

Importantly, our discrimination components operate only during training, leaving the inference architecture unchanged. This design ensures no additional computational overhead during deployment—a critical advantage for real-world applications where efficiency

is paramount.

The following subsection provides the formal mathematical framework that underpins our multi-granularity discrimination approach, establishing the foundation for the detailed technical descriptions in subsequent sections.

3.2.6 Mathematical Formulation

Formally, given an untrimmed video sequence V , we extract video clips $x \in \mathcal{X}$ through sliding window sampling from both action video segments and non-action video segments. Each action clip x is associated with an action category label $y \in \mathcal{Y}$, while non-action clips are used for discrimination learning without requiring explicit action labels.

A standard action recognition model f_{θ_A} learns a set of parameters θ_A to predict the action label y from input clip x by optimizing the cross-entropy loss. To enhance the model’s discriminative capability through our multi-granularity framework, we decompose the network architecture into two functional components:

$$f_{\theta_A} = h_{\theta_C} \circ g_{\theta_B} \quad (3.1)$$

where $\theta_A = \theta_B \cup \theta_C$. In this formulation:

- g_{θ_B} (depicted as pink cubes in Figure 3.3) represents the **feature extraction function** learned by the early layers of the model (typically consisting of three to four convolutional or transformer blocks). This component extracts low-to-mid-level spatiotemporal features.
- h_{θ_C} (depicted as blue cubes in Figure 3.3) represents the **classification function** learned by the remaining deeper layers of the model. This component performs high-level reasoning and final action classification.

Our objective is to enhance both θ_B and θ_C by learning auxiliary discrimination tasks that inject discriminative information into both shallow and deep layers. This multi-level approach results in stronger feature representations that focus on temporal movement patterns rather than static contextual cues.

To achieve this temporal discrimination learning, we propose three complementary auxiliary loss functions in addition to the standard classification loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_1 \mathcal{L}_{\text{BL}} + \lambda_2 \mathcal{L}_{\text{CL}} + \lambda_3 \mathcal{L}_{\text{CML}} \quad (3.2)$$

where each loss component corresponds to one level of our discrimination framework:

- $\mathcal{L}_{\text{CE}}$ is the standard cross-entropy loss for action classification on action clips
- $\mathcal{L}_{\text{BL}}$ is the **Booster Network Loss** that facilitates coarse-level discrimination between action and non-action clips across different videos
- $\mathcal{L}_{\text{CL}}$ is the **Contrastive Learning Loss** that enhances fine-level discrimination between action and non-action clips within the same video sequence
- $\mathcal{L}_{\text{CML}}$ is the **Clip Matching Loss** that enables finer-level discrimination between clips sampled from different temporal positions within segments
- λ_1 , λ_2 , and λ_3 are hyperparameters that balance the contribution of each auxiliary loss component

Each auxiliary loss function targets a specific granularity level of temporal discrimination, collectively enhancing the model’s ability to learn discriminative features that are robust to variations in viewpoint, execution style, and background context. The multi-task learning framework allows the model to simultaneously optimize for multiple discriminative objectives, progressively refining its understanding from coarse inter-video differences to fine intra-segment temporal dynamics.

This mathematical framework provides the foundation for implementing our coarse-to-fine-to-finer discrimination strategy. In the following sections, we provide detailed explanations of each loss component, their specific implementations, and their contributions to the overall discriminative learning framework.

3.3 Coarse Level: Discrimination Across Videos

This section presents the first level of our hierarchical action recognition framework, which focuses on learning to discriminate between action and non-action clips across different videos. This coarse-level discrimination serves as the foundation for our approach, establishing initial action-aware features in the early layers of the network that can be progressively refined in subsequent stages.

3.3.1 Motivation and Problem Definition

As established in our framework overview, **coarse-level discrimination** operates across different videos with diverse visual contexts. The goal is to enable the model to distinguish between action clips (sampled from action video segments) and non-action clips (sampled from non-action video segments) regardless of background variations, scene settings, or visual contexts.

Through extensive analysis of action recognition datasets, we observe that several categories of untrimmed videos commonly coexist in similar scenarios. For example, as shown in Figure 3.2(a), multiple types of activities often occur in the same environment, such as a playground or sports field where both “throwing frisbee” (action) and “meddling with frisbee” (non-action) can occur. This situation is prevalent across many action recognition datasets, where the visual context alone provides insufficient information to distinguish between different activities.

When examining video clips depicting different actions in similar environments, we recognize an opportunity: comparing action clips with non-action clips across different videos allows us to acquire distinct discriminative information that transcends individual video contexts. This cross-video comparison helps the model gradually learn to disregard scene-specific information and focus on the temporal dynamics that characterize actions, regardless of their visual context.

An untrimmed video typically contains both action video segments (containing the actions of interest) and non-action video segments (containing background activities, transitions, or preparatory movements). Video clips sampled from these different segment types often exhibit high semantic similarity in their spatial appearance, making discrimination based solely on static visual cues extremely challenging. To effectively identify actions, the model must learn to recognize distinctive motion patterns and temporal dynamics rather than relying on static scene elements.

3.3.2 Booster Networks: Definition and Design

To achieve effective coarse-level discrimination, we propose attaching auxiliary sub-networks to the early layers of the backbone model. These auxiliary networks are designed to differentiate between action and non-action clips, thereby guiding the feature extraction process toward more discriminative representations. We refer to these auxiliary sub-networks as “**booster networks**” (or “**booster nets**”) since they enhance or “boost” the corresponding layers with discriminative information.

Booster Network Definition: A booster network is a lightweight auxiliary classifier that takes intermediate feature representations from early backbone layers as input and outputs binary predictions distinguishing action clips from non-action clips. The key characteristics of booster networks include:

- **Auxiliary nature:** They operate in parallel to the main classification pathway without disrupting the primary action recognition task
- **Early layer attachment:** They are connected to early layers (typically the first 3-4 blocks) to inject discriminative signals early in the feature extraction process
- **Binary classification:** They perform action/non-action discrimination rather than multi-class action classification
- **Training-only operation:** They are used only during training and removed during inference, ensuring no additional computational overhead

The booster nets can be implemented using various architectural choices, depending on the specific task requirements and the level of information to be propagated to the early layers of the model. Since our goal at this coarse level is to learn basic discriminative information between action and non-action clips across videos, we employ a straightforward approach: simple feedforward linear layers that predict binary action/non-action labels. This lightweight design ensures that the auxiliary task does not overwhelm the main action recognition objective while still providing valuable discriminative signals.

3.3.3 Integration with Backbone Architecture

We denote the booster networks by $f_{\theta_b}^j$, where θ_b represents the parameters of the auxiliary network and j represents the index of the booster network. These booster networks are attached to the first n blocks or layers of the backbone model, represented as $g_{\theta_B}^j$ for $j = 1$ to n , where n is chosen empirically based on the backbone architecture (we use $n = 3$ in our experiments based on ablation studies).

A key advantage of our approach is that the auxiliary network design is agnostic to the specific backbone architecture. This flexibility allows us to apply our method to any 2D or 3D action recognition backbone, including:

- CNN-based architectures (TSM-ResNet, I3D, SlowFast)

- Transformer-based models (TimeSformer, VideoSwin)
- Hybrid architectures (TANet, TPN)

The integration process simply involves tapping into the intermediate feature maps produced by the early layers of the backbone and feeding them into the corresponding booster networks.

3.3.4 Training Objective and Loss Function

Let $\ell_i \in \{0, 1\}$ denote the binary label indicating whether an input video clip x_i belongs to the “action” class ($\ell_i = 1$) or “non-action” class ($\ell_i = 0$). To train the booster networks, we employ the standard binary cross-entropy loss to penalize inaccurate action/non-action predictions.

The booster network loss function $\mathcal{L}_{\text{BL}}$ (as defined in Equation 2) is formulated as:

$$\mathcal{L}_{\text{BL}} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^n [\ell_i \log p_j^i + (1 - \ell_i) \log(1 - p_j^i)] \quad (3.3)$$

where:

- N is the batch size
- n is the number of booster networks (equal to the number of early layers they are attached to)
- $p_j^i = f_{\theta_b}^j(g_{\theta_B}^j(x_i))$ represents the predicted probability that clip x_i is an action clip, as determined by the j -th booster network
- $g_{\theta_B}^j(x_i)$ represents the feature map produced by the j -th layer of the backbone model when processing input clip x_i
- ℓ_i is the ground truth binary label for clip x_i

By optimizing for this objective, the early layers of the backbone model become increasingly action-aware. This awareness is crucial because it enables the model to start distinguishing between action and non-action clips from the earliest stages of processing, establishing a foundation for more refined discrimination in subsequent layers.

3.3.5 Information Flow and Feature Propagation

The action-aware discriminative features learned through the booster networks are propagated to the later layers of the model through the standard feed-forward process. This propagation serves two critical purposes:

1. **Highlighting action cues:** The early discrimination helps emphasize the temporal and motion patterns that characterize action clips while de-emphasizing static contextual elements.
2. **Suppressing static context:** By focusing on action-specific features, the model learns to downweight the static contextual elements that co-occur with actions but do not define them, effectively addressing the shortcut learning problem.

This propagation of discriminative information from early to later layers creates a cascading effect that progressively refines the model’s understanding of actions. The coarse-level discrimination across videos provides a broad foundation of action awareness, upon which more fine-grained discrimination (as discussed in Section 3.4) can be built.

By establishing this hierarchical flow of information, the model becomes increasingly capable of distinguishing between action and non-action clips, even when they share similar visual contexts. This coarse-to-fine approach mirrors the human perceptual process, where broad categorical distinctions precede fine-grained discrimination, resulting in more effective and interpretable action recognition.

3.4 Fine Level: Discrimination Within a Video

This section explores the second level of our hierarchical action recognition framework, which focuses on discriminating between action and non-action clips within the same video sequence. As established in our framework overview, **fine-level discrimination** operates within the same video context, where action clips and non-action clips share identical visual elements—the same subjects, objects, backgrounds, and scenes—but differ in their temporal activities.

This fine-grained discrimination is particularly challenging yet crucial, as it enables the model to distinguish between temporally similar activities that share identical visual contexts, subjects, and objects. For example, differentiating between clips containing “throwing a frisbee” versus clips containing “holding a frisbee” requires understanding subtle

temporal patterns rather than simply recognizing objects or scenes. As illustrated in Figure 3.2(b), this level forces the model to focus on movement dynamics while suppressing static contextual information that remains constant across both clip types.

We begin by reviewing relevant contrastive learning methods that form the foundation of our approach, then present our novel contrastive learning technique specifically designed for within-video discrimination.

3.4.1 Background: Contrastive Learning

Contrastive learning has emerged as a powerful paradigm in representation learning, offering effective ways to learn discriminative features by comparing similar and dissimilar examples. This section reviews key contrastive learning frameworks that inform our approach to fine-level action discrimination, providing the theoretical foundation for our within-video contrastive learning method.

Contrastive learning methods typically aim to pull semantically similar samples closer in the embedding space while pushing dissimilar samples apart. Several prominent approaches have been developed in the self-supervised learning literature. Wu et al. [110] pioneered this direction by introducing a memory bank to store computed instance representations, employing noise contrastive estimation to distinguish between instance and noise representations. This approach treats each instance as its own class, enabling unsupervised representation learning without requiring explicit labels during training.

MoCo [35] significantly enhanced this approach by implementing a momentum encoder to maintain a dynamic dictionary of representations. This momentum update mechanism allows for a large and consistent dictionary of negative examples without requiring massive batch sizes, achieving remarkable success in various vision tasks. The key insight is that the momentum encoder provides more stable negative examples by slowly updating the encoder weights, leading to more consistent training dynamics.

SimCLR [17] proposed an alternative method using large mini-batches instead of a memory bank, demonstrating that with sufficient batch size and appropriate data augmentations, simple contrastive learning can achieve competitive performance without the need for specialized architectures. SimCLR’s success highlighted the importance of negative sampling and data augmentation strategies in contrastive learning.

When extended to video domains, specialized contrastive learning approaches have been developed to account for the temporal dimension. CoCLR [34] takes advantage of the inherent multi-modal nature of videos by leveraging the natural correlations between RGB

and optical flow modalities to identify positive samples. This cross-modal approach helps the model learn more robust representations that capture both appearance and motion information, which is particularly relevant for action recognition tasks.

VideoMoCo [75] further adapts contrastive learning for videos by applying temporal decay on the “key” representations in the memory queue and employing a generator to strategically discard multiple frames, creating a more temporally resilient encoder that can better handle variations in video timing and pacing. This temporal adaptation addresses the unique challenges of video representation learning.

InfoNCE

Self-supervised video representation learning aims to produce a feature generator G that can be utilized to effectively encode input data for different downstream applications. To understand the foundation of our approach, we first examine the InfoNCE loss, which serves as the basis for many contrastive learning methods and forms the theoretical backbone of our within-video discrimination approach.

In the InfoNCE framework, we define positive pairs $\mathcal{P}$ as the augmentation ψ of data $x_i \in X_i$, where a represents a specific augmentation sampled from the augmentation distribution A . The augmentation distribution typically includes operations such as random cropping, color jittering, and temporal sampling for videos. The negative pairs $\mathcal{N}$ comprise augmented versions of all other instances in the dataset.

Formally, we define the positive and negative pairs as:

$$\mathcal{P}_i = \{\psi(x_i; a) | a \sim A\} \quad (3.4)$$

$$\mathcal{N}_i = \{\psi(x_n; a) | \forall n \neq i, a \sim A\} \quad (3.5)$$

The InfoNCE [17] loss function is then expressed as:

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E} \left[\log \frac{\exp(z_i \cdot z_p / \tau)}{\exp(z_i \cdot z_p / \tau) + \sum_{n \in \mathcal{N}_i} \exp(z_i \cdot z_n / \tau)} \right] \quad (3.6)$$

where $z_i = G(\psi(x_i; \cdot))$ represents the encoded feature of the augmented instance, τ is a temperature parameter that controls the sharpness of the distribution, and $z_i \cdot z_p$ represents the dot product measuring similarity between two feature vectors. This loss function encourages the model to maximize the agreement between differently augmented views of the same instance while minimizing similarity with other instances.

UberNCE

When we have access to class labels y_i for data $x_i \in X_i$, the standard InfoNCE approach can be modified to leverage this supervisory signal. The key insight here is that InfoNCE’s strict instance discrimination might be unnecessarily restrictive for certain tasks, particularly when instances of the same class should have similar representations.

UberNCE [34] extends the contrastive learning framework by redefining positive pairs to include augmentations of all samples from the same class, rather than just the sample itself, while maintaining the overall structure of the contrastive loss. This class-aware approach allows for more effective representation learning when categorical information is available, which is particularly relevant for our supervised action recognition setting.

The positive and negative pairs in UberNCE are redefined as:

$$\begin{aligned}\mathcal{P}_i &= \{\psi(x_i; a), x_p | y_p = y_i \text{ and } p \neq i, \forall p \in [1, N], a \sim A\} \\ \mathcal{N}_i &= \{\psi(x_n; a), x_n | y_n \neq y_i, \forall n \in [1, N], a \sim A\}\end{aligned}\tag{3.7}$$

This formulation treats all instances of the same class as positive pairs, which is particularly useful in supervised settings where we want representations to capture class-level semantics rather than purely instance-level distinctions. This class-aware contrastive learning serves as an important foundation for our approach.

3.4.2 Proposed Fine-Level Contrastive Learning

Building upon the contrastive learning foundations discussed above, we now present our novel approach for fine-grained discrimination within videos. Our key observation is that within the same video sequence, action clips and non-action clips frequently share identical visual elements—the same subjects, objects, backgrounds, and scenes where the actions occur. This presents both a challenge and an opportunity: while these shared elements make discrimination difficult, they also allow us to focus specifically on learning the temporal patterns that distinguish actions from non-actions in otherwise identical contexts.

To develop fine-grained discriminative features for a particular untrimmed video sequence, we propose learning auxiliary tasks in the later layers of the neural network model to enhance action representations. These auxiliary tasks are designed to force the model to focus on the temporal dynamics that differentiate action clips from non-action clips, rather than relying on visual appearance cues that remain constant across both clip types.

A naive approach might be to use a triplet loss function to differentiate between action and non-action clips. In such an approach, one might consider another action clip from the same video as a positive sample and non-action clips as negative samples. However, this approach presents practical challenges: it requires large numbers of negative samples for effective training, which becomes computationally prohibitive with video data. Due to memory and processing constraints, we simply cannot include all video clips in a single input batch, which limits the performance of this method and may even cause performance deterioration. Moreover, such auxiliary tasks can interfere negatively with the primary task of action classification, reducing overall accuracy. We have empirically verified these limitations, and the experimental results are presented in Section 3.6.4 Table 3.7.

Inspired by recent advances in contrastive learning, particularly SimCLR [17] and MoCo [35], we propose an extension of the contrastive loss from UberNCE [34] to learn fine-grained representations in the penultimate layer of the action recognition model. Our approach also introduces a novel sampling strategy specifically designed for selecting positive and negative pairs within the same video for the contrastive learning objective.

Video Clip Feature Store

To address the computational challenges associated with processing multiple video clips simultaneously, we introduce a **video clip feature store** mechanism. This approach allows us to effectively utilize a large number of negative samples without incurring prohibitive computational costs during training.

Feature Store Definition: The video clip feature store is a memory bank that contains pre-computed feature representations for all clips sampled from a given untrimmed video. These features are extracted from the penultimate layer of the backbone model and stored in chronological order, preserving temporal relationships within the video sequence.

We first create a video clip feature store containing concatenated clip features extracted for the entire video from the penultimate layer of a pre-trained action recognition model. These features are saved in chronological order, preserving the temporal relationships within the video. The feature store serves as a memory bank of representations that can be efficiently accessed during training without needing to recompute features for each clip multiple times.

For each video V , we create a feature store $\mathcal{F}_V = \{f_1, f_2, \dots, f_T\}$ where f_t represents the penultimate layer feature of the t -th clip in the video, and T is the total number of clips sampled from video V . This mechanism provides several advantages:

- **Computational efficiency:** Avoids recomputing features for the same clips multiple times during training
- **Large negative sampling:** Enables access to many negative samples without memory constraints during batch processing
- **Temporal preservation:** Maintains the chronological order of clips within the video sequence
- **Scalability:** Allows processing of long videos with hundreds of clips efficiently

Contrastive Loss for Within-Video Discrimination

Using the video clip feature store, we apply a contrastive loss to learn discriminative information between the current input clip’s penultimate feature f^p and other clip features in our store. Specifically, we sample penultimate clip features from action clips within the same video as positive samples, while using all clip features from non-action clips as negative samples.

This contrastive learning objective serves as an auxiliary task that complements the primary action recognition objective. It compels the model to learn subtle temporal differences between action and non-action clips while minimizing interference from subjects, objects, and scenes that remain constant across the video. By focusing specifically on what differentiates action from non-action in identical visual contexts, the model develops a more refined understanding of action-specific temporal patterns.

Let $\mathcal{P}$ denote positive pairs and $\mathcal{N}$ denote negative pairs, representing penultimate features from the model. For our within-video contrastive learning, we redefine the positive and negative pair sampling strategy to focus on temporal discrimination within the same video context.

Let $\ell_i \in \{0, 1\}$ denote the binary label indicating whether clip x_i is an action clip ($\ell_i = 1$) or non-action clip ($\ell_i = 0$). The positive and negative pairs for our within-video contrastive loss are defined as:

$$\begin{aligned}\mathcal{P}_i &= \{f_j \in \mathcal{F}_V | \ell_j = \ell_i \text{ and } j \neq i\} \\ \mathcal{N}_i &= \{f_j \in \mathcal{F}_V | \ell_j \neq \ell_i\}\end{aligned}\tag{3.8}$$

where $f_j = h_{\theta_C}^p(g_{\theta_B}(x_j))$ represents the penultimate layer feature of clip x_j , ℓ_i indicates whether clip i contains action or non-action content, and all clips belong to the same video

V . The crucial distinction here is that all clips come from the same video V , ensuring that our contrastive learning focuses on temporal differences rather than visual context differences.

Projection and Loss Computation

To project the feature layer f^p to a dedicated contrastive representation space, we employ a **projection layer** $f_{\theta_{\text{proj}}}^{\text{proj}}$ consisting of a simple feedforward network with two linear layers and a ReLU activation between them. This projection helps separate the representation space used for the main action recognition task from the space used for contrastive learning, reducing potential interference between tasks.

Projection Layer Purpose: The projection layer serves several critical functions in our framework:

- **Task separation:** Creates a dedicated representation space for contrastive learning, independent from the main classification space
- **Interference reduction:** Minimizes negative interactions between the main action recognition task and the auxiliary contrastive task
- **Representation optimization:** Enables learning of representations specifically optimized for temporal discrimination
- **Dimensionality control:** Allows control over the dimensionality of the contrastive learning space

The fine-level contrastive learning loss $\mathcal{L}_{\text{CL}}$ (as defined in Equation 2) is computed on the projected features f^{proj} as follows:

$$\mathcal{L}_{\text{CL}} = -\mathbb{E} \left[\log \frac{\exp(z_i \cdot z_p / \tau)}{\exp(z_i \cdot z_p / \tau) + \sum_{f_n \in \mathcal{N}_i} \exp(z_i \cdot z_n / \tau)} \right] \quad (3.9)$$

Here, $z_i = f_{\theta_{\text{proj}}}^{\text{proj}}(h_{\theta_C}^p(g_{\theta_B}(x_i)))$ represents the projected feature of the input clip, τ represents the temperature parameter for scaling the features, and $z_i \cdot z_p$ represents the similarity (dot product) between two feature vectors. This loss encourages the model to pull representations of action clips closer together while pushing them away from non-action clips, even when they share identical visual contexts.

By incorporating this fine-level contrastive learning objective as an auxiliary task during training, our model learns to focus on the distinctive temporal patterns that characterize specific actions, rather than being distracted by shared visual elements that remain constant across both action and non-action clips. This approach enables more accurate discrimination within videos, addressing one of the core challenges in temporal action localization and directly tackling the shortcut learning problem identified in current action recognition systems.

The discriminative features learned through this fine-level contrastive learning are propagated to subsequent layers, providing enhanced representations for even more fine-grained discrimination as discussed in Section 3.5. This progressive refinement from coarse inter-video discrimination to fine intra-video discrimination creates increasingly sophisticated temporal understanding capabilities within our hierarchical framework.

3.5 Finer Level: Discrimination Inside One Segment

This section presents the third and finest level of our hierarchical discrimination framework, which focuses on **finer-level discrimination within individual segments**. As established in our framework overview, this level operates within individual action or non-action segments to detect subtle differences between clips sampled from different temporal positions within the same segment. Unlike the previous levels that addressed discrimination across videos (Section 3.3) and within videos (Section 3.4), finer-level discrimination captures the internal temporal structure and dynamics that characterize how actions unfold over time.

As illustrated in Figure 3.2(c), this finest level of discrimination enables precise localization of action boundaries and helps the model focus on spatio-temporal variations in moving subjects rather than static scene information, addressing the most subtle aspects of temporal understanding.

3.5.1 Motivation and Challenges

We observe that even within a single action or non-action segment, clips acquired by dense sampling at different starting points exhibit a high degree of visual similarity. They typically share identical scenes, subjects, and objects, but differ subtly in their motion patterns and temporal progression. This presents both a challenge and an opportunity:

by learning to recognize these subtle temporal variations, we can enhance the model’s awareness of motion changes within the temporal scale of individual segments.

Traditional action recognition approaches often treat segments as uniform entities, failing to capture the internal temporal structure that characterizes how actions unfold over time. For instance, a ”throwing” action comprises a sequence of distinct sub-movements (preparation, execution, and follow-through), each with unique temporal signatures that might be critical for distinguishing it from similar actions. By developing sensitivity to these internal variations, our model can achieve more precise temporal understanding and localization of actions.

3.5.2 Background: Cross-Modal Alignment Approaches

Our approach to finer-level discrimination draws inspiration from recent advances in cross-modal alignment techniques, particularly those used in vision-language models. These methods have demonstrated remarkable abilities to align representations across different modalities, suggesting potential applications for temporal alignment within video understanding.

CLIP and Cross-Modal Alignment

Contrastive Language-Image Pre-training (CLIP) [83] introduced a powerful framework for aligning visual and textual representations. CLIP trains on a large dataset of image-text pairs, encouraging text representations to closely match their corresponding image representations in a shared embedding space. This is achieved through a contrastive loss that maximizes similarity between matching pairs while minimizing similarity between non-matching pairs.

The key insight from CLIP that informs our approach is its ability to create alignments between different types of data by measuring and optimizing their similarity in a shared representation space. While CLIP focuses on cross-modal alignment between images and text, we adapt this principle to the temporal domain, aligning clip-level features with their temporal context within the full video sequence.

Other Alignment Approaches

Besides CLIP, several other approaches have explored alignment in multimedia contexts. For instance, MIL-NCE [73] proposed a multiple instance learning framework for aligning

narrations with corresponding video segments. ALIGN [47] demonstrated that noisy web-scale data could be leveraged for image-text alignment through a contrastive approach. These works highlight the versatility of similarity-based alignment techniques and their potential for application in temporal contexts.

From Cross-Modal to Temporal Alignment

The key insight we draw from these cross-modal alignment approaches is their ability to establish correspondences between different types of representations through similarity computation. In our temporal context, we adapt this principle to align individual clip features with their temporal positions within complete video sequences. Instead of aligning text with images, we align clips with their temporal context, enabling the model to understand where specific clips belong within the broader temporal structure of actions.

This temporal alignment approach allows our model to develop fine-grained understanding of how actions progress through time, capturing the subtle temporal dynamics that distinguish between different phases of action execution within individual segments.

3.5.3 Proposed Clip Matching Mechanism

Clip Matching Definition: Clip matching is a temporal localization mechanism that learns to identify the temporal position of individual clips within complete video sequences through similarity computation. The core objective is to enable the model to understand the internal temporal structure of segments by learning to distinguish between clips sampled from different temporal positions within the same segment.

Inspired by cross-modal alignment approaches, we hypothesize that locating the position of input video clips on the temporal timeline will help the model develop finer-level discriminative capabilities within individual segments. We design a similarity matching network, which we call "clip matching," to locate the penultimate feature of an input video clip within our saved video clip feature store with temporal order.

Given the challenging nature of precise clip localization in action videos with high visual similarity, we set a more achievable objective: we want the model to at least localize whether a clip falls within or outside the action segment. This binary discrimination encourages the model to become more sensitive to the distinctive features that characterize action segments versus non-action segments, even when they share identical visual contexts.

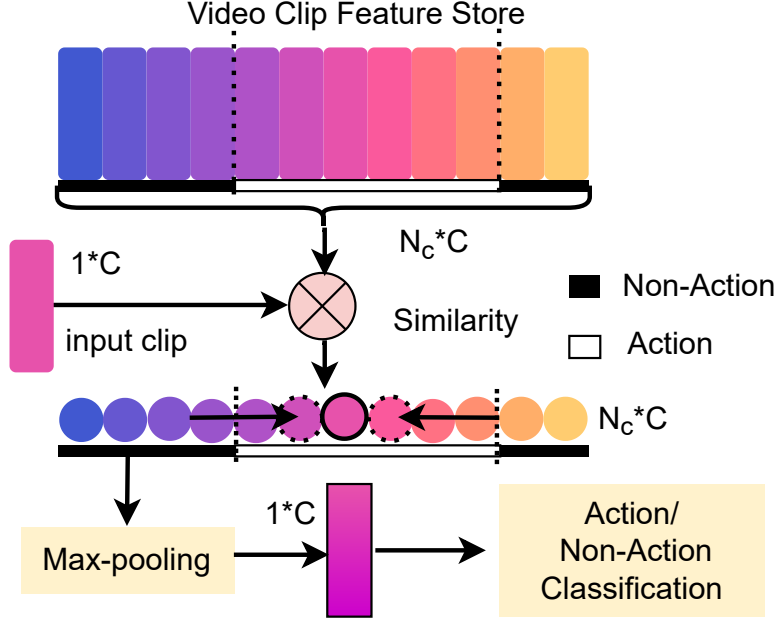


Figure 3.4: Illustration of the clip matching mechanism. The model computes similarity between a clip’s features and all temporal positions in the video feature store, then identifies the most likely temporal position through max-pooling. This encourages the model to learn temporally discriminative features that can precisely locate a clip within the video timeline.

3.5.4 Clip to Video-Clip Similarity Matching

The clip matching procedure represents a critical component of our coarse-to-fine-to-finer discriminative learning framework, enabling the model to establish correspondences between individual clips and their contextual positions within complete video sequences. This mechanism addresses the fundamental challenge of understanding how isolated action segments relate to their broader temporal context, which is essential for robust action recognition in untrimmed videos. The procedure operates as illustrated in Figure 3.4 and consists of several interconnected steps that progressively refine the matching process.

Feature Extraction and Normalization

The initial stage involves extracting rich feature representations from both individual clips and complete video sequences using a pre-trained action recognition backbone. Let $f^p(x_i) \in$

$\mathbb{R}^C$ denote the penultimate features (features from the layer immediately before the final classification layer) output from an action recognition model for an input clip x_i , where C represents the dimensionality of the penultimate feature space. These penultimate features are particularly valuable as they contain high-level semantic information while maintaining sufficient discriminative power for matching operations.

For an entire video X with temporal length N_c (representing the number of clips obtained through temporal segmentation), containing clips x_i for $i = 1 \dots N_c$, the complete video feature representation is constructed through temporal concatenation:

$$f_G^p(X) := ||_{i=1}^{N_c} f^p(x_i) \in \mathbb{R}^{N_c \times C} \quad (3.10)$$

where $||$ denotes the concatenation operation along the temporal dimension, and $f_G^p(X)$ represents the global video features containing the sequential arrangement of all clip-level features. This representation preserves the temporal structure of the video while maintaining the semantic richness of individual clip features.

To enable effective similarity comparison across different temporal positions and mitigate the effects of feature magnitude variations, we apply L2 normalization to the video features:

$$f_{G_N}^p(X) = \frac{f_G^p(X)}{||f_G^p(X)||_2} \quad (3.11)$$

where $f_{G_N}^p(X) \in \mathbb{R}^{N_c \times C}$ represents the normalized video features, ensuring consistent comparison magnitudes across different temporal positions and reducing the impact of feature scale variations that could bias the matching process.

Similarity Computation and Temporal Localization

The core matching mechanism employs element-wise similarity computation to identify correspondences between the input clip and potential temporal positions within the video. We compute the similarity matrix between the normalized video features and the individual clip feature $f^p(x_i)$ using the Hadamard product (element-wise multiplication):

$$sim(f_{G_N}^p, f_i^p) = f_{G_N}^p(X) \odot f_i^p(x_i) \in \mathbb{R}^{N_c \times C} \quad (3.12)$$

where $\odot$ represents the Hadamard product operation, and the resulting similarity matrix captures the element-wise correspondence between the input clip features and each temporal position in the video. This operation effectively measures how well each feature dimension of the input clip aligns with the corresponding dimensions at different temporal locations.

To identify the most discriminative temporal position and extract the strongest matching signal, we apply max-pooling along the temporal dimension. This operation identifies the maximally activated response for each feature channel across all temporal positions:

$$f_{max} = \max_{\{t=1,2,\dots,N_c\}} (sim(f_{G_N}^p, f_i^p)) \in \mathbb{R}^C \quad (3.13)$$

The max-pooling operation serves multiple purposes: it identifies the temporal position with the strongest feature correspondence, provides translation invariance across temporal positions, and creates a compact representation that captures the best matching characteristics between the clip and video.

The resulting feature map f_{max} is then processed through a Sigmoid Linear Unit (SiLU) activation function [24]:

$$f_{CML}(x_i; X) = \text{SiLU}(f_{max}) = f_{max} \cdot \sigma(f_{max}) \quad (3.14)$$

where $\sigma(\cdot)$ represents the sigmoid function. The SiLU activation provides several advantages: it offers self-gating properties that allow the network to selectively emphasize important features, provides smooth gradients that facilitate stable training, and has been shown to improve convergence properties in deep networks compared to traditional ReLU activations.

Classification and Loss Computation

The activated feature representations are subsequently processed through a feedforward classification layer that produces binary predictions indicating whether the input clip corresponds to an action or non-action segment within the video context:

$$p_{CML} = \text{Linear}(f_{CML}(x_i; X)) \in \mathbb{R}^2 \quad (3.15)$$

where $\text{Linear}(\cdot)$ represents a fully connected layer that maps the SiLU-activated features to class logits.

The training objective employs binary cross-entropy loss to optimize the clip matching predictions:

$$L_{CML} = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^2 \ell_{i,k} \log(\text{softmax}(p_{CML})_k) \quad (3.16)$$

where N is the batch size, $\ell_{i,k}$ represents the one-hot encoded ground truth label for sample i and class k , and $\text{softmax}(\cdot)$ ensures proper probability normalization. The loss function

encourages the model to learn discriminative features that can effectively distinguish between action and non-action clips based on their contextual similarity with complete video sequences.

This clip matching loss L_{CML} corresponds to the finer-level component in our total loss function, working in conjunction with the coarse-level booster loss L_{BL} and fine-level contrastive loss L_{CL} to create a comprehensive multi-granularity discrimination framework.

3.5.5 Integration and Advanced Analysis

Multi-Level Feature Fusion

The clip matching module integrates seamlessly with the coarse and fine-level discrimination components through our comprehensive loss function framework. This integration allows the model to leverage discriminative information from all three granularity levels simultaneously during training, creating more robust and comprehensive action representations.

Temporal Sensitivity Analysis

The clip matching mechanism enhances temporal sensitivity by forcing the model to recognize subtle variations in motion patterns within segments. Through ablation studies, we observe that models trained with clip matching show improved sensitivity to:

- **Action boundaries:** Better localization of action start and end points within longer video sequences, enabling more precise temporal segmentation
- **Motion transitions:** Enhanced recognition of sub-action phases and transitions between different movement components within complex actions
- **Temporal consistency:** Improved understanding of action progression and the natural flow of movements over time
- **Fine-grained dynamics:** Increased sensitivity to subtle temporal variations that distinguish similar actions or different execution styles

Computational Efficiency Considerations

Despite operating on complete video sequences, the clip matching mechanism maintains computational efficiency through several design choices:

- **Feature reuse:** Pre-computed features stored in the feature store eliminate redundant computation, as features are computed once and reused for multiple matching operations
- **Efficient similarity computation:** Element-wise operations enable parallel processing and can be efficiently implemented using modern GPU architectures
- **Training-only overhead:** The matching component operates only during training, ensuring no additional computational cost during inference deployment
- **Scalable architecture:** The mechanism scales linearly with video length, making it applicable to videos of varying durations

Robustness to Temporal Variations

The max-pooling operation in our similarity computation provides robustness to temporal variations commonly encountered in real-world videos:

- **Speed variations:** Actions performed at different speeds are handled through temporal alignment, as the matching mechanism can identify corresponding temporal phases regardless of execution speed
- **Timing shifts:** Small temporal misalignments are compensated by the matching mechanism’s ability to find the best correspondence across the temporal sequence
- **Duration differences:** Variable action durations are accommodated through adaptive matching that focuses on the most discriminative temporal features
- **Execution style variations:** Different styles of performing the same action are handled through the robust feature matching that emphasizes core temporal patterns while being tolerant to stylistic variations

3.5.6 Integration with Overall Framework

The clip matching mechanism serves as the third level in our hierarchical discrimination framework, focusing on fine-grained temporal localization within segments. By combining this with the coarse-level discrimination across videos (Section 3.3) and fine-level discrimination within videos (Section 3.4), we create a comprehensive approach that addresses action recognition at multiple granularities.

Our final loss function integrates all three levels of discrimination:

$$L = \lambda_{CE}L_{CE} + \lambda_{BL}L_{BL} + \lambda_{CL}L_{CL} + \lambda_{CML}L_{CML} \quad (3.17)$$

where:

- L_{CE} is the primary cross-entropy loss for action classification
- L_{BL} is the booster loss for coarse-level discrimination (Section 3.3)
- L_{CL} is the contrastive loss for fine-level discrimination (Section 3.4)
- L_{CML} is the clip matching loss for finer-level discrimination (this section)
- λ_{CE} , λ_{BL} , λ_{CL} , and λ_{CML} are hyperparameters that weight the different loss terms

In our experiments, we empirically set these hyperparameters to $\lambda_{CE} = \lambda_{BL} = \lambda_{CML} = 1$ and $\lambda_{CL} = 0.5$ to balance the contributions of each loss term.

3.5.7 Benefits of Finer-Level Discrimination

The clip matching mechanism provides several important benefits for action recognition:

1. **Temporal Awareness:** By encouraging the model to locate clips within the video timeline, it develops enhanced sensitivity to temporal progression and dynamics.
2. **Sub-action Understanding:** The model becomes more aware of the internal structure of actions, potentially recognizing sub-actions or phases within a complete action.

3. **Precise Localization:** This mechanism improves the model’s ability to precisely localize the boundaries of action segments, which is crucial for temporal action detection tasks.
4. **Motion-Focused Features:** By focusing on temporal positioning rather than static visual elements, the model learns to prioritize motion patterns over scene context.

Through this third level of discrimination, our model develops a comprehensive understanding of actions that spans from coarse cross-video distinctions to fine-grained temporal awareness within individual segments. This hierarchical approach mirrors human perception, where understanding progresses from broad categorical distinctions to fine-grained temporal sequences, resulting in more robust and accurate action recognition.

3.6 Experiments and Results

This section presents a comprehensive evaluation of our proposed hierarchical discrimination approach. We begin by introducing the datasets used for evaluation, then detail our implementation methodology, and finally present extensive experimental results that demonstrate the efficacy of our approach across various action recognition backbones. We also conduct thorough ablation studies to analyze the contribution of each component in our framework.

3.6.1 Datasets

To evaluate the effectiveness of our approach, we conducted experiments on three widely-used video action datasets, each with distinct characteristics and challenges. All three datasets provide annotations for both action and non-action segments, which is essential for our discrimination-based framework.

ActivityNet-v1.3

ActivityNet-v1.3 [9] stands as one of the most comprehensive benchmarks for video understanding, supporting multiple tasks including action recognition, temporal action localization, action proposal generation, and dense video captioning. The dataset consists of 10,024 training videos and 4,926 validation videos, all of which are untrimmed and annotated with temporal boundaries for 200 distinct action categories.

The videos in ActivityNet-v1.3 are sourced from online platforms and represent a wide variety of daily activities, ranging from personal care and household activities to sports and socializing. This diversity makes the dataset particularly challenging and representative of real-world scenarios.

A key characteristic of ActivityNet-v1.3 is the relatively long average action segment duration of 117 seconds, which allows for the analysis of extended temporal dynamics within actions. The non-action segments in these videos often contain contextually related activities, making the discrimination between action and non-action particularly challenging and relevant for our approach.

HACS

The Human Action Clips and Segments (HACS) dataset [120] represents a significant scaling up of action recognition data, containing approximately 139,000 action segments meticulously annotated in 50,000 untrimmed videos across 200 action categories. The scale and diversity of HACS make it particularly suitable for training deep learning models that require substantial amounts of data.

HACS was constructed with a focus on temporal action localization, ensuring precise temporal boundaries for each action segment. Each action segment is constrained to be under 4 minutes in length, with an average duration of 2.6 minutes. This makes HACS somewhat similar to ActivityNet in terms of segment duration, but with a significantly larger number of samples, providing greater statistical power for model evaluation.

The 200 action categories in HACS are aligned with those in ActivityNet, facilitating cross-dataset comparisons and enabling researchers to leverage both datasets in complementary ways. The non-action portions in HACS videos often contain related contextual information, presenting similar discrimination challenges as ActivityNet.

FineAction

The FineAction dataset [64] represents a more fine-grained approach to action detection, containing approximately 103,000 action segments across 106 categories, annotated in 17,000 untrimmed videos. What distinguishes FineAction from the previous two datasets is its focus on shorter, more precise action segments, with an average duration of just 7.1 seconds.

This shorter average duration presents a distinct challenge compared to ActivityNet and HACS, as the model must learn to recognize actions with less temporal context. The

fine-grained nature of the annotations in FineAction also requires models to develop more precise temporal discrimination abilities, making it particularly suitable for evaluating the finer-level discrimination capabilities of our approach.

The 106 action categories in FineAction encompass a wide range of human activities, including many that require subtle distinctions between visually similar actions. This characteristic aligns well with our focus on learning discriminative features that can distinguish between action and non-action segments based on temporal dynamics rather than static visual cues.

3.6.2 Implementation Details

All experiments were implemented using PyTorch on Tesla V100 GPUs. To thoroughly validate our hierarchical discrimination approach, we integrated our framework with several state-of-the-art action recognition architectures, demonstrating its versatility and effectiveness across different backbone designs.

For our contrastive learning and clip matching components, we constructed a video feature store by extracting 2048-dimensional vectors from the penultimate layer of a pre-trained TSM-ResNet50 [57] model. Features were uniformly sampled across each video, producing approximately 100 temporal feature vectors per video. The projection layer used in contrastive learning reduces these features to a 128-dimensional embedding space through a two-layer MLP with ReLU activation.

Below, we detail the specific implementation for each backbone architecture:

TSN We implemented TSN [101] with ResNet50 [36] as backbone, sampling 8 frames with a frame interval of 1 for each segment. Frames were resized to 256 pixels on the shorter side and randomly cropped to 224×224 during training. The model was fine-tuned for 25 epochs on 2 Tesla V100 GPUs with a batch size of 16, using SGD optimization with an initial learning rate of 1e-3. The booster networks were attached after the first three ResNet blocks.

TSM For TSM [57], we used a ResNet50 backbone enhanced with temporal shift operations. Input frames were resized to 308×308 and randomly cropped to 224×224. We sampled 8 frames with a frame interval of 8 for each action segment. Training ran for 25 epochs on 2 Tesla V100 GPUs with a batch size of 16, using SGD optimization with a learning rate of 5e-5. Our booster networks were integrated after each temporal shift

module in the first three blocks, leveraging the temporal modeling already present in the architecture.

TPN The TPN [116] architecture was implemented with a ResNet50 backbone, with inputs resized directly to 224×224 pixels. For each clip, we sampled 8 frames with a frame interval of 8. Training consisted of 150 epochs on 2 Tesla V100 GPUs with a batch size of 64, using SGD with an initial learning rate of $2.5e-3$. We attached booster networks at multiple levels of the temporal pyramid, enabling discrimination at different temporal scales.

TANet For TANet [69], frames were resized to 256 pixels on the shorter side and cropped to 224×224 . We sampled 8 frames with a frame interval of 1, focusing on fine-grained temporal dynamics. The model was fine-tuned for 50 epochs on 2 GPUs using SGD with a learning rate of $1e-3$. Our booster networks were integrated after each temporal aggregation module in the first three blocks.

TimeSformer For the Transformer-based TimeSformer [6], inputs were rescaled to 320×256 pixels and randomly cropped to 224×224 . We sampled 8 frames with a frame interval of 32, allowing the model to capture long-range temporal dependencies. Fine-tuning ran for 15 epochs on 2 GPUs with a batch size of 16, using SGD with a learning rate of $1e-4$. We adapted our booster networks to operate on the output of the first three Transformer blocks.

Our final loss function combines four components:

$$L = \lambda_{CE}L_{CE} + \lambda_{BL}L_{BL} + \lambda_{CL}L_{CL} + \lambda_{CML}L_{CML} \quad (3.18)$$

where $\lambda_{CE} = \lambda_{BL} = \lambda_{CML} = 1$ and $\lambda_{CL} = 0.5$.

During evaluation, we uniformly sampled 10 clips per video and averaged their predictions for the final classification. Performance was evaluated using standard metrics: top-1 and top-5 accuracy for action recognition, and mean average precision (mAP) at different temporal IoU thresholds for temporal action localization.

3.6.3 Results

Our proposed framework was comprehensively evaluated across multiple state-of-the-art video classification backbone architectures on three benchmark datasets: ActivityNet,

HACS, and FineAction. The experimental results consistently demonstrate that our approach enhances the performance of existing models, regardless of their architectural design or underlying principles.

Evaluation Protocol

We employed top-1 single crop accuracy as our primary evaluation metric, which is the standard approach in trimmed action recognition tasks. Unlike previous studies that rely solely on video-level action labels, our framework leverages untrimmed non-action data by utilizing both action and action/non-action labels for computing accuracy metrics. Importantly, our evaluation focuses exclusively on action segments, providing a more refined assessment of model performance.

This methodological distinction is crucial to consider when comparing our results with those reported in other literature. Many published works report significantly higher accuracy figures because they use multi-crop video-level predictions, which artificially inflate performance metrics by averaging predictions across multiple spatial crops and temporal segments. In contrast, our single crop, action segment-level evaluation presents a more challenging and realistic assessment of model performance.

Furthermore, most studies in the literature report only video-level accuracy, which typically aggregates predictions across multiple temporal segments. While our primary focus is on the more granular segment-level analysis, we recognized the need to provide direct comparisons with existing methods. Therefore, to validate our approach’s effectiveness in terms comparable with the literature, we conducted additional video-level experiments (shown in Table 3.4). In these experiments, we achieved 94.8% accuracy on the ActivityNet 2022 challenge, outperforming other state-of-the-art methods. These video-level results serve as additional evidence of our framework’s effectiveness when evaluated using the same metrics as prior work, while our segment-level analysis remains our principal and more detailed evaluation approach.

It is worth emphasizing that the absolute accuracy values reported in our tables should not be directly compared with those in papers using different evaluation protocols. Our results, while potentially showing lower absolute numbers, represent a more stringent and faithful measure of action recognition performance, and the consistent improvements across various backbones validate the effectiveness of our approach under these rigorous conditions.

¹denotes the results we obtained using the official public code TSM [57] and mmaction2 [19].

Table 3.1: Comprehensive performance comparisons of state-of-the-art architectures on the ActivityNet benchmark dataset. The architectures include ResNet (RN), Vision Transformer base version (ViT-B), and Swin Transformer base version (Swin-B). Performance metrics are reported across multiple input configurations. #F denotes the number of input frames processed by each model, where c indicates video clips (each clip consists of 8 consecutive frames). Acc represents the top-1 classification accuracy measured on action clips. Models were trained using the standard ActivityNet training protocol with cross-entropy loss and evaluated on the validation set. All experiments were conducted using identical data augmentation techniques and optimization hyperparameters to ensure fair comparison.

Method	Pre-train	Backbone	ActivityNet	
			#F	Acc
TSN ¹	Kinetics	RN50	25	75.4
TSN+ours	Kinetics	RN50	25	78.5
TSM ¹	Kinetics	RN50	8	75.8
TSM+ours	Kinetics	RN50	8	77.7
TPN ¹	Kinetics	RN50	8	72.9
TPN+Ours	Kinetics	RN50	8	75.7
TANet ¹	Kinetics	RN50	8	76.9
TANet+Ours	Kinetics	RN50	8	79.9
Timesformer ¹	Kinetics	VIT-B	8	83.5
Timesformer+Ours	Kinetics	VIT-B	8	84.5
VideoSwin ¹	Kinetics	Swin-B	32	86.7
VideoSwin+Ours	Kinetics	Swin-B	32	87.9

Performance on ActivityNet

Table 3.1 presents a comprehensive comparison of our framework against baseline models on the ActivityNet dataset. Our approach consistently enhanced performance across all tested backbone architectures, ranging from traditional CNN-based models like ResNet-50 to more recent transformer-based architectures.

The integration of our framework with TSN yielded a substantial improvement of 3.1 percentage points, increasing the accuracy from 75.4% to 78.5%. Similarly, when applied to TSM, our method improved performance by 1.9 percentage points (75.8% to 77.7%).

The most significant improvement was observed with TPN, where our framework boosted accuracy by 2.8 percentage points, from 72.9% to 75.7%.

For TANet, the incorporation of our framework resulted in a notable 3.0 percentage point improvement, increasing accuracy from 76.9% to 79.9%. This substantial enhancement highlights our method’s ability to effectively complement temporal attention mechanisms.

When applied to transformer-based architectures, our framework continued to demonstrate its efficacy. For Timesformer, we observed a 1.0 percentage point improvement (83.5% to 84.5%), while VideoSwin with our framework achieved an impressive 87.9% accuracy, representing a 1.2 percentage point increase over the baseline. It is worth noting that these transformer-based models already exhibit strong baseline performance, making the improvements even more significant.

Table 3.2: Comprehensive performance comparisons of state-of-the-art architectures on the HACS benchmark dataset. #F denotes the number of input frames processed by each model. Acc represents the top-1 classification accuracy measured on action clips. Models were trained using the standard HACS training protocol with cross-entropy loss and evaluated on the validation set. All experiments were conducted using identical data augmentation techniques and optimization hyperparameters to ensure fair comparison.

Method	Back bone	HACS	
		#F	Acc
TSM ¹	RN50	8	82.45
TSM+Ours	RN50	8	84.21
TANet ¹	RN50	8	88.79
TANet+Ours	RN50	8	89.13
Timesformer ¹	VIT-B	8	89.87
Timesformer+Ours	VIT-B	8	90.21

Performance on HACS

The results on the HACS dataset, presented in Table 3.2, further corroborate the effectiveness of our approach. When integrated with TSM, our framework improved accuracy

¹denotes the results we obtained using the official public code TSM [57] and mmaction2 [19].

by 1.76 percentage points, from 82.45% to 84.21%. For TANet, we observed a modest but meaningful improvement of 0.34 percentage points (88.79% to 89.13%).

The integration with Timesformer yielded a 0.34 percentage point improvement, increasing accuracy from 89.87% to 90.21%. The relatively smaller improvements observed on HACs compared to ActivityNet can be attributed to the already high baseline accuracies, which may constrain the potential for further enhancement. Additionally, the use of TSM features for contrastive learning and clip matching may not have been optimal for other architectural designs, potentially explaining the more modest improvements for certain models.

Table 3.3: Comprehensive performance comparisons of state-of-the-art architectures on the FineAction benchmark dataset. #F denotes the number of input frames processed by each model. Acc represents the top-1 classification accuracy measured on action clips. Models were trained using the standard FineAction training protocol with cross-entropy loss and evaluated on the validation set. All experiments were conducted using identical data augmentation techniques and optimization hyperparameters to ensure fair comparison.

Method	Backbone	#F	Acc
TSM ¹	RN50	8	67.18
TSM+Ours	RN50	8	71.18
TSN ¹	RN50	25	69.25
TSN+Ours	RN50	25	71.45
TANet ¹	RN50	8	67.98
TANet+Ours	RN50	8	70.93
Timesformer ¹	VIT-B	8	73.56
Timesformer+Ours	VIT-B	8	73.89

Performance on FineAction

Table 3.3 demonstrates our framework’s versatility across different video classification backbones on the FineAction dataset. The most substantial improvement was observed with TSM, where our approach enhanced accuracy by 4.0 percentage points (67.18% to 71.18%). This significant improvement highlights our method’s effectiveness in addressing the fine-grained action recognition challenges presented by this dataset.

¹denotes the results we obtained using the official public code TSM [57] and mmaction2 [19].

For TSN, our framework improved performance by 2.2 percentage points (69.25% to 71.45%), while TANet with our approach achieved a 2.95 percentage point improvement (67.98% to 70.93%). Even for the already high-performing Timesformer, our method yielded a modest but consistent improvement of 0.33 percentage points (73.56% to 73.89%).

These results on FineAction underscore our framework’s ability to enhance action recognition performance across diverse backbone architectures, particularly in challenging scenarios involving fine-grained action categories.

Table 3.4: Video level result on ActivityNet. Performance comparison of our multi-crop strategy with SOTA methods on the ActivityNet dataset.

method	Top-1 Acc
MARL[107]	85.7
TSQNet[111]	88.7
NSNet[112]	90.2
Text4Vis[108]	93.3
BIKE ViT-L[109]	94.7
Ours	94.8%

Video-Level Performance in ActivityNet Challenge

In addition to the segment-level evaluations, we assessed our framework’s effectiveness at the video level during the ActivityNet 2022 challenge. As shown in Table 3.4, our multi-crop strategy achieved a remarkable 94.8% top-1 video-level accuracy, outperforming several state-of-the-art approaches. This result represents a marginal improvement over the previous best performer, BIKE ViT-L [109], which achieved 94.7% accuracy.

The competitive performance in this challenge further validates the robustness and efficacy of our proposed framework in real-world settings, demonstrating its practical value for high-performance video understanding applications.

Analysis of Performance Improvements

The consistent performance enhancements observed across different architectures and datasets can be attributed to several key aspects of our framework:

Architectural Synergy: Our method demonstrates varying degrees of synergy with different backbone architectures. The most substantial improvements were observed with CNNs like TSM and TANet, suggesting that our approach effectively complements the spatio-temporal feature extraction capabilities of these models. The more modest improvements with transformer-based architectures may be attributed to their inherently stronger modeling of long-range dependencies, an aspect our framework specifically enhances.

Dataset Characteristics: The magnitude of improvement varies across datasets, with ActivityNet showing the most substantial gains. This variation can be attributed to the different characteristics of each dataset, including the diversity of action categories, video duration, and inherent challenges such as background clutter or viewpoint variations.

Baseline Performance: Models with lower baseline performance generally exhibited larger improvements when integrated with our framework. This pattern suggests that our approach effectively addresses certain limitations in these models, potentially related to their ability to distinguish between action and non-action segments or to capture fine-grained temporal dynamics.

Feature Compatibility: The use of TSM features for contrastive learning and clip matching may have limited the potential improvements for other architectural designs. This observation highlights the importance of feature compatibility in our framework and suggests potential for further enhancement through architecture-specific feature extraction strategies.

In summary, our comprehensive evaluation across multiple datasets and backbone architectures demonstrates the versatility and effectiveness of our proposed framework in enhancing video classification performance. The consistent improvements observed, regardless of the underlying model architecture or dataset characteristics, underscore the robustness of our approach and its potential for broad applicability in various video understanding tasks.

3.6.4 Ablation Study

To rigorously evaluate the contribution of individual components within our proposed framework, we conducted a comprehensive ablation study using the ActivityNet-v1.3 dataset with TSM [57] as the backbone architecture. This analysis provides valuable insights into

Table 3.5: Different loss components on ActivityNet. This table presents the impact of our proposed loss functions on the TSM baseline model. We evaluate combinations of three components: inter-video Booster, intra-video Contrastive, and intra-segment Clip Matching losses. Top-1 accuracy is reported on ActivityNet-v1.3. The baseline achieves 75.84%, while the full model with all three components achieves the best performance at 77.71%.

Baseline	Booster (inter-video)	Contrastive (intra-video)	Clip Matching (intra-segment)	ActivityNet Accuracy
✓				75.84
✓	✓			77.10
✓		✓		77.46
✓			✓	77.45
✓	✓	✓		77.63
✓		✓	✓	77.68
✓	✓		✓	77.56
✓	✓	✓	✓	77.71

how each component contributes to the overall performance improvement and their synergistic interactions.

Analysis of Different Loss Components

Table 3.5 presents the performance impact of different combinations of our proposed loss functions. The baseline TSM model achieves 75.84% top-1 accuracy on ActivityNet-v1.3. When we systematically integrate our proposed loss components, we observe consistent and substantial improvements across all configurations.

Individual Component Analysis: When applied individually, each component demonstrates significant effectiveness:

- The inter-video Booster component improves the baseline by 1.26 percentage points (75.84% $\rightarrow$ 77.10%), highlighting its effectiveness in enhancing discriminative learning across different videos.

- The intra-video Contrastive loss component yields the highest individual improvement of 1.62 percentage points (75.84% $\rightarrow$ 77.46%), suggesting the importance of learning discriminative features within the same video.
- The intra-segment Clip Matching component provides a comparable improvement of 1.61 percentage points (75.84% $\rightarrow$ 77.45%), demonstrating the value of aligning features at the clip level within segments.

These results confirm that each component addresses a distinct aspect of action recognition challenges, with the Contrastive and Clip Matching components showing particularly strong individual contributions.

Pairwise Component Integration: When combining components in pairs, we observe further performance enhancements:

- Booster + Contrastive yields 77.63% accuracy, a 1.79 percentage point improvement over the baseline.
- Contrastive + Clip Matching achieves the highest pairwise performance at 77.68%, a 1.84 percentage point gain over the baseline.
- Booster + Clip Matching results in 77.56% accuracy, a 1.72 percentage point improvement.

The superior performance of the Contrastive + Clip Matching combination suggests a particularly strong complementarity between intra-video and intra-segment learning objectives. This synergy can be attributed to their focus on different granularities within the same video: the Contrastive component optimizes feature discrimination across different segments, while the Clip Matching component refines temporal alignment within segments.

Full Framework Performance: When all three components are integrated, our framework achieves the highest performance of 77.71%, representing a substantial 1.87 percentage point improvement over the baseline. This result demonstrates that despite the strong individual and pairwise contributions, the full combination still provides additional gains, albeit marginal compared to the best pairwise combination (0.03 percentage points over Contrastive + Clip Matching).

Hierarchical Learning Analysis: The observed performance patterns reveal a hierarchical discriminative learning process operating at multiple scales:

- The Booster component operates at the coarsest level, focusing on inter-video discrimination to distinguish between different action categories across videos.
- The Contrastive component functions at a finer level, enhancing intra-video discrimination between action and non-action segments within the same video.
- The Clip Matching component works at the finest level, optimizing intra-segment temporal relationships between clips.

This multi-scale hierarchical approach enables our framework to learn discriminative features at different granularities, progressively refining the model’s ability to distinguish action from non-action content and recognize specific action categories.

Component Complementarity: The consistent performance improvements across different component combinations, with the best results achieved when all components are used, confirm our hypothesis that these components target complementary aspects of action recognition. This complementarity is particularly evident in how the relative improvement from adding a third component diminishes compared to the gains from the first and second components, suggesting that while there is some overlap in the information captured, each component still contributes unique discriminative signals.

In conclusion, our ablation study demonstrates that each component of our proposed framework makes meaningful and distinct contributions to the overall performance. The hierarchical nature of these components—operating at inter-video, intra-video, and intra-segment levels—creates a coarse-to-fine-to-finer discriminative learning paradigm that effectively enhances action recognition capabilities. This comprehensive analysis validates our design choices and provides valuable insights into the mechanisms underlying our framework’s effectiveness.

Analysis of Positive and Negative Pairs Selection

The effectiveness of contrastive learning is heavily influenced by the strategy used for selecting positive and negative pairs. To investigate this critical aspect, we conducted a series of experiments examining different sampling strategies for negative pairs in our

Table 3.6: Ablation on the positive and negative pairs selection methods for contrastive learning on ActivityNet-v1.3 dataset. We compare different negative sampling strategies: using only the sampled video itself (1 video), using videos from the same action class (628 videos), and a hybrid approach combining both (629 videos).

Method	Acc
Baseline	75.84
Sampled video features(1 video)	77.46
Same action class video features(628 videos)	76.86
Sampled video + same action class features(629 videos)	77.26

contrastive learning framework. Table 3.6 presents the results of these experiments on the ActivityNet-v1.3 dataset.

Our baseline TSM model achieves 75.84% accuracy. When we implement our contrastive learning approach using features from only the sampled video itself (1 video) for negative pair generation, we observe a substantial improvement to 77.46%, representing a 1.62 percentage point gain over the baseline. This configuration yields the highest performance among all tested strategies.

When expanding the negative pair sampling to include features from videos of the same action class (628 videos), the performance drops to 76.86%. While this still represents a meaningful improvement over the baseline (+1.02 percentage points), it is 0.60 percentage points lower than using only the sampled video features. This performance degradation suggests that incorporating features from different videos of the same action class introduces confounding factors that may interfere with the model’s ability to learn fine-grained action discriminative features.

We further investigated a hybrid approach that combines both sampling strategies, using features from both the sampled video and other videos of the same action class (629 videos in total). This approach achieves 77.26% accuracy, which is better than using only same-action class videos (+0.40 percentage points) but still underperforms compared to using only the sampled video (-0.20 percentage points).

These results reveal several important insights about contrastive learning for action recognition:

Intra-video Discrimination: The superior performance of using only the sampled video features demonstrates the critical importance of learning to distinguish variations within the same video. This intra-video discrimination appears to be more valuable than cross-video discrimination within the same action class. This finding aligns with the inherent challenge in action recognition: distinguishing between action and non-action segments within the same video context.

Context Confounds: When features from multiple videos are used for negative pair generation, the model may inadvertently learn to discriminate based on scene context or background differences rather than action-specific motion patterns. Different videos, even of the same action class, often contain substantial variations in background, lighting, camera angle, and other contextual factors. These variations may overshadow the subtle distinctions between action and non-action segments that are crucial for effective action recognition.

Feature Space Complexity: The hybrid approach’s intermediate performance suggests that while incorporating a broader range of negative examples can provide some benefit, it also increases the complexity of the feature space the model must navigate. The slight performance decrease compared to using only the sampled video indicates that the added complexity may partially outweigh the benefits of increased sample diversity.

Implications for Contrastive Learning Design: These findings have important implications for designing effective contrastive learning frameworks for action recognition. They suggest that focusing on intra-video distinctions may be more effective than attempting to learn cross-video distinctions, particularly when the goal is to distinguish between action and non-action segments.

This ablation study provides empirical evidence supporting our design decision to focus primarily on intra-video feature discrimination in our contrastive learning component. By prioritizing the ability to distinguish between different segments within the same video, our framework effectively learns the subtle temporal and motion patterns that differentiate actions from non-actions in a consistent visual context, leading to superior action recognition performance.

Table 3.7: Comparison of our method with other intuitive background awareness methods on ActivityNet-v1.3 dataset. We evaluate five alternative approaches: (1) adding a global background class, (2) creating action-specific background classes, (3) basic triplet loss with intra-video negatives, (4) expanded triplet loss with batch-level negatives, and (5) feature-based triplet loss. Results demonstrate how different strategies for incorporating non-action (background) information affect the action recognition performance compared to our proposed method.

Models	Acc
Baseline TSM	75.84
200 actions + 1 background	66.99
200 actions + 200 backgrounds	56.36
Triplet loss (1 positive 2 negative)	71.88
Triplet loss (1 positive 2 negative) in one batch	73.49
Triplet loss on features	77.37
Our method	77.71

Comparative Analysis of Background Awareness Methods

Incorporating non-action (background) information effectively is a critical challenge in action recognition systems. While the presence of non-action segments in untrimmed videos provides potentially valuable contextual information, the optimal way to leverage this information is not immediately obvious. We conducted a systematic comparison of several intuitive approaches for incorporating background awareness, evaluating their impact on action recognition performance on the ActivityNet-v1.3 dataset. Table 3.7 presents the results of this comparative analysis.

Alternative Approaches: We investigated five alternative methods for incorporating background awareness:

1. **Adding a global background class** (200 actions + 1 background): This approach treats all non-action segments as a single additional class, expanding the original 200 action categories to 201 classes.
2. **Action-specific background classes** (200 actions + 200 backgrounds): This more

fine-grained approach creates a separate background class for each action category, resulting in 400 total classes.

3. **Basic triplet loss** (1 positive, 2 negative): This method samples one positive action clip and two negative non-action clips from the same video and applies triplet loss to enforce feature separation.
4. **Expanded triplet loss** (1 positive, 2 negative in one batch): This extends the basic triplet loss by considering negative samples from all videos within a batch.
5. **Feature-based triplet loss**: This approach applies triplet loss directly on feature representations rather than on raw video clips.

Performance Analysis: The baseline TSM model achieves 75.84% accuracy on ActivityNet-v1.3. Our experimental results reveal striking differences in the effectiveness of the various background awareness methods:

- Methods 1 and 2, which explicitly model background as additional classes, show substantial performance degradation. The global background class approach (Method 1) reduces accuracy to 66.99% (-8.85 percentage points), while the action-specific background approach (Method 2) performs even worse at 56.36% (-19.48 percentage points). This significant performance drop suggests that treating background segments as explicit classes creates fundamental challenges for the classification model.
- Methods 3 and 4, which employ triplet loss on video clips, also underperform the baseline. The basic triplet loss (Method 3) achieves only 71.88% accuracy (-3.96 percentage points), while the expanded triplet loss (Method 4) performs somewhat better at 73.49% (-2.35 percentage points), but still falls short of the baseline.
- Method 5, which applies triplet loss on feature representations, is the only alternative approach that improves upon the baseline, achieving 77.37% accuracy (+1.53 percentage points). This indicates that operating at the feature level rather than the raw clip level is crucial for effective background-aware learning.
- Our proposed method outperforms all alternatives, achieving 77.71% accuracy (+1.87 percentage points over the baseline and +0.34 percentage points over the best alternative). This demonstrates the superiority of our approach in effectively leveraging background information.

Theoretical Implications: These results offer several important insights into the nature of background-aware action recognition:

- **Classification vs. Discrimination:** The poor performance of class-based approaches (Methods 1 and 2) suggests that framing background awareness as a classification problem is suboptimal. Background segments are inherently diverse and lack the coherent visual patterns that define action classes, making them unsuitable for standard classification approaches.
- **Feature-Level Learning:** The superior performance of feature-based methods (Method 5 and our approach) indicates that background awareness is more effectively incorporated at the feature representation level rather than through explicit classification. This allows the model to learn the relationship between action and non-action segments in a more nuanced manner.
- **Loss Function Design:** The performance gap between our method and the feature-based triplet loss (Method 5) highlights the importance of choosing an appropriate contrastive learning objective. Our temperature-scaled cross-entropy loss appears better suited to the action recognition context than triplet loss, possibly due to its ability to consider multiple negative samples simultaneously and its more stable gradient properties during training.

Methodological Implications: The comparative analysis provides strong empirical validation for our design choices. The poor performance of seemingly intuitive approaches, such as adding explicit background classes or applying triplet loss on video clips, underscores the non-trivial nature of effectively incorporating background awareness. Our approach, which focuses on feature-level discrimination using a carefully designed contrastive learning objective, represents a more sophisticated and effective solution to this challenge.

The marginal but consistent improvement of our method over the feature-based triplet loss (+0.34 percentage points) further demonstrates the importance of specific implementation details in contrastive learning frameworks. Our temperature-scaled cross-entropy loss appears to provide better optimization properties and more effective feature separation compared to triplet loss, leading to superior action recognition performance.

In conclusion, this comparative analysis not only validates the effectiveness of our proposed approach but also provides valuable insights into the fundamental challenges of background-aware action recognition. By systematically evaluating alternative methods

Table 3.8: Ablation study of using deeper layers and attention mechanism for clip matching on ActivityNet-v1.3 dataset. We compare three architectural variants: a simple single linear layer, a two-layer feedforward network (FFN), and an FFN augmented with Squeeze-and-Excitation (SE) attention.

Linear layer	FFN	SE Block	Accuracy
✓			77.71
	✓		77.58
	✓	✓	77.34

and analyzing their performance characteristics, we establish a stronger theoretical foundation for understanding why our approach works and how it differs from more intuitive but less effective alternatives.

Analysis of Clip Matching Architecture

Different Components for Clip Matching: To determine the optimal architecture for our clip matching module, we conducted an ablation study examining the effects of using different network components. Table 3.8 presents the performance impact of incorporating deeper network layers and attention mechanisms on the ActivityNet-v1.3 dataset.

Our experimental results demonstrate that the simpler architecture using a single linear layer achieves the best performance at 77.71% accuracy. When we replace this with a more complex feedforward network (FFN) with two layers, the performance decreases slightly to 77.58%, representing a 0.13 percentage point reduction. This performance degradation suggests that the additional parameters introduced by the deeper network may lead to overfitting on the clip matching task, indicating that a single linear layer provides sufficient modeling capacity for this component.

Further augmenting the architecture by adding a Squeeze-and-Excitation (SE) attention block [38] to the feedforward network results in a further performance decrease to 77.34%, which is 0.37 percentage points lower than the single linear layer configuration. This more substantial drop indicates that the SE attention mechanism is not well-suited for the clip matching task in our framework. The attention mechanism may be focusing on features that are less relevant for distinguishing between action and non-action segments, potentially diluting the discriminative power of the learned representations.

Table 3.9: Ablation studies on the number of booster nets on ActivityNet-v1.3 dataset. We test with 1-4 booster nets and compare binary (action/non-action) discrimination with N-way classification approaches.

Method	Accuracy
Baseline TSM	75.84
1-layer Booster	77.36
2-layer Booster	77.53
3-layer Booster	77.71
4-layer Booster	77.45
N-classification 3-layer Booster	76.87

These findings align with the principle of parsimony in model design—simpler architectures often generalize better when the modeling capacity of more complex alternatives exceeds what is required for the task. In our case, the clip matching component appears to function optimally with minimal architectural complexity, allowing it to effectively capture the essential discriminative features without overfitting to training data idiosyncrasies.

Analysis of Booster Network Architecture

Variations in Booster Network Depth: Table 3.9 presents results of varying booster network depth on ActivityNet-v1.3. The baseline TSM achieves 75.84% accuracy, while adding a single-layer booster network increases performance to 77.36% (+1.52 percentage points). Performance continues to improve with 2-layer (77.53%) and 3-layer configurations (77.71%), demonstrating that cascaded action/non-action discrimination at multiple network depths enhances feature representations. However, extending to 4 layers decreases performance to 77.45%, indicating an optimal depth exists beyond which additional layers become detrimental. This decline likely occurs because too many booster networks dominate the optimization process, compromising the backbone’s ability to learn optimal representations for fine-grained action classification.

Binary vs. N-way Classification: We also evaluated an N-way classification variant that directly predicts action categories, rather than our binary action/non-action approach. This N-classification with a 3-layer booster architecture achieves 76.87% accuracy, which is lower than our binary approach (-0.84 percentage points) while still outperforming the baseline. This result validates our design decision to focus on binary discrimination, which

Table 3.10: Comparison of action localization results on ActivityNet dataset using BMN [58] model using the features extracted from the model trained using our proposed method against the baseline without using our method. This experiment evaluates the transferability of our action/non-action discriminative learning approach to the downstream task of temporal action localization. Results are reported using mean Average Precision (mAP) at different temporal Intersection over Union (IoU) thresholds (0.5, 0.75, 0.95) and their average.

Features (BMN model)	0.5	0.75	0.95	avg mAP
TSM	50.55	35.33	6.52	34.15
TSM+Ours	50.80	35.29	6.83	34.32
Timesformer	46.16	30.94	4.38	29.94
Timesformer+Ours	46.40	30.67	4.07	30.14

provides a more effective form of coarse-level discriminative learning that complements the final classification task by better separating action from non-action content.

Temporal Action Localization Evaluation

Action Localization Performance: To further validate the effectiveness of our discriminative learning approach, we evaluated its impact on the downstream task of temporal action localization. This experiment aims to determine whether the enhanced discriminative capabilities between action and non-action segments translate to improved boundary detection in untrimmed videos. We extracted features from models trained with our proposed method and utilized them with the Boundary Matching Network (BMN) [58], a state-of-the-art action localization framework.

Table 3.10 presents the localization performance on the ActivityNet dataset using features extracted from both baseline models and their counterparts enhanced with our method. For the TSM backbone, our approach improves the average mAP from 34.15% to 34.32%, representing a 0.17 percentage point gain. Examining performance across different IoU thresholds reveals that our method provides the most substantial improvements at IoU=0.5 (50.55% $\rightarrow$ 50.80%) and IoU=0.95 (6.52% $\rightarrow$ 6.83%), with a slight decrease at IoU=0.75 (35.33% $\rightarrow$ 35.29%). The enhanced performance at the highest IoU threshold (0.95) is particularly noteworthy, as it indicates that our method enables more precise boundary localization.

For the Timesformer backbone, we observe similar trends with an average mAP improvement from 29.94% to 30.14% (+0.20 percentage points). The improvement is most pronounced at IoU=0.5 (46.16% $\rightarrow$ 46.40%), while performance at IoU=0.75 shows a small decrease (30.94% $\rightarrow$ 30.67%) and IoU=0.95 also exhibits a slight reduction (4.38% $\rightarrow$ 4.07%).

These results demonstrate that the discriminative features learned by our method enhance the model’s ability to distinguish action boundaries, particularly at lower and higher IoU thresholds. The improvements in localization performance, though modest, provide additional evidence of our approach’s effectiveness in learning representations that better differentiate between action and non-action segments. This enhanced discrimination capability directly translates to more accurate action proposals for localization, validating the broader applicability of our method beyond the primary task of action recognition.

Feature Activation Visualization

To provide qualitative insights into the mechanisms underlying our framework’s effectiveness, we visualized the feature activation maps using Grad-CAM [84]. These visualizations help us understand how our method alters the network’s attention patterns during action recognition.

Figure 3.5 presents the feature activation visualizations across multiple action categories, including doing karate, windsurfing, swimming, and playing water polo. A notable observation is that our method consistently focuses on the regions containing action-relevant motion, rather than the surrounding scene context. This is particularly evident in categories with potential scene bias, where standard models might rely heavily on background cues rather than action-specific features.

The comparative visualization in Figure 3.6 further illustrates the difference between our approach and the original method. Our framework demonstrates enhanced focus on action-relevant regions while reducing attention on irrelevant objects in the scene. This shift in attention patterns provides visual confirmation of our method’s ability to prioritize action-discriminative features, supporting our quantitative findings on improved action recognition performance.

These visualizations offer compelling evidence that our discriminative learning approach successfully guides the model to attend to action-relevant cues while suppressing background information, resulting in more robust action representations that are less susceptible to dataset biases.

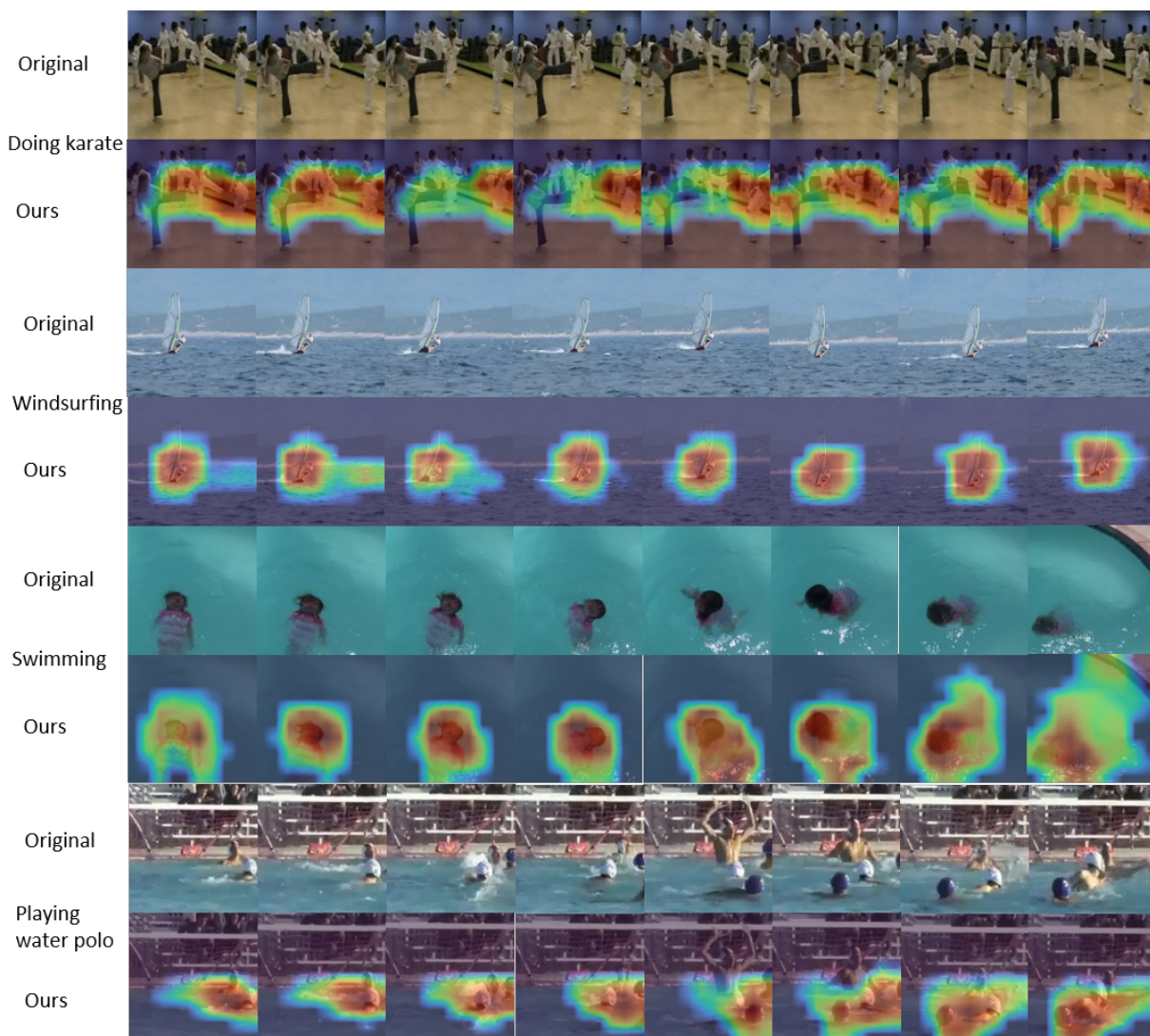


Figure 3.5: Visualization of feature activation maps using GRAD-CAM across multiple action categories, including doing karate, windsurfing, swimming, and playing water polo. A notable observation is that our method consistently focuses on the regions containing action-relevant motion, rather than the surrounding scene context. This is particularly evident in categories with potential scene bias, where standard models might rely heavily on background cues rather than action-specific features.

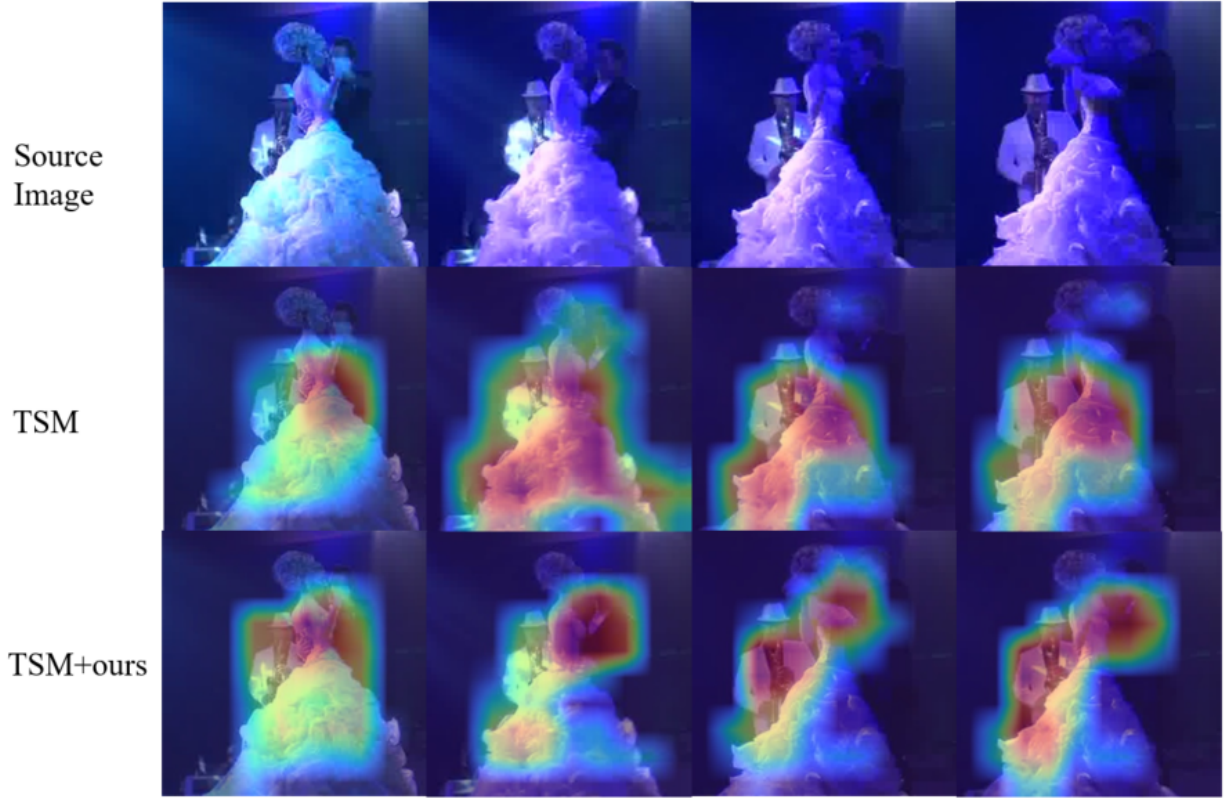


Figure 3.6: The comparative visualization of feature activation graphs between our method and the original method using GRAD-CAM. Our framework demonstrates enhanced focus on action-relevant regions while reducing attention on irrelevant objects in the scene. This shift in attention patterns provides visual confirmation of our method’s ability to prioritize action-discriminative features, supporting our quantitative findings on improved action recognition performance.

Cross-Dataset Generalization to Trimmed Videos

While our framework was primarily designed for untrimmed videos that contain both action and non-action segments, a critical question remains: Can our approach generalize to trimmed datasets where all frames contain action content? To address this question, we conducted experiments on the UCF-101 dataset, a widely-used benchmark for trimmed action recognition.

The fundamental challenge in applying our method to trimmed datasets is the absence

of non-action segments that serve as negative examples in our contrastive learning framework. To overcome this limitation, we implemented a modified version of our approach using only the booster layers, with non-action clips from ActivityNet videos serving as external negative samples. This adaptation demonstrates an important aspect of our framework’s flexibility: the ability to leverage negative samples from different sources when they are not available within the target dataset itself.

Our experimental results show an average improvement of **0.4%** across the three standard test splits of the UCF-101 dataset compared to the baseline. Although this improvement is modest, its consistency across all splits provides compelling evidence that the core principles of our approach can effectively generalize beyond untrimmed videos.

This successful cross-dataset application is particularly significant because it demonstrates that the action/non-action discriminative learning paradigm can enhance performance even in datasets where traditional approaches might assume it offers no advantage. By incorporating external non-action samples, our framework effectively transfers the discriminative knowledge learned from untrimmed videos to improve feature representations in trimmed action recognition tasks.

These findings expand the potential application scope of our method beyond its original untrimmed video context, suggesting that the principles of action/non-action discriminative learning may have broader implications for video understanding tasks, regardless of whether the target videos themselves contain non-action content.

3.7 Chapter Summary and Conclusion

This chapter presented a comprehensive analysis of our proposed coarse-to-fine-to-finer discriminative learning framework for action recognition, establishing the foundational level of our multi-level temporal understanding approach. We demonstrated that leveraging non-action segments from untrimmed videos can significantly enhance action recognition performance through a multi-level discriminative learning approach. Our framework introduces three complementary components: (1) a generic booster mechanism for coarse-level inter-video discrimination, (2) a strategically designed contrastive learning approach for fine-level intra-video discrimination, and (3) a clip matching method for finer-level intra-segment discrimination.

Through extensive experiments on ActivityNet, HACS, and FineAction datasets, we validated that our approach consistently improves performance across diverse backbone architectures, including both CNN-based models like TSM and transformer-based models

like Timesformer. Our ablation studies provided valuable insights into the contribution of each component, the optimal architectural choices, and the mechanisms underlying our framework’s effectiveness. The visualization analysis further confirmed that our method successfully guides models to focus on action-relevant regions rather than background context, addressing a common limitation in existing approaches.

Importantly, we demonstrated the versatility of our framework through experiments on trimmed videos and action localization tasks, showing that the principles of action/non-action discrimination can generalize beyond their original context. The model-agnostic nature of our approach, combined with its zero additional inference cost, makes it a practical solution for enhancing action recognition systems in real-world applications.

The consistent improvements observed across diverse settings confirm our central hypothesis that explicitly modeling the distinction between action and non-action content through progressive discriminative learning can substantially enhance a model’s ability to recognize actions, even in challenging and biased scenarios. By focusing on what makes actions distinctive from non-actions, our approach establishes robust temporal feature representations that capture the essential dynamics of human activities.

While our framework advances the capability to recognize actions in video segments (addressing the ”what” question in our multi-level understanding framework), many real-world scenarios involve untrimmed videos where actions occur at unknown temporal locations. This presents the challenge of temporal action localization—determining not only what actions are present but precisely when they occur (the ”when” question). In Chapter 4, we address this challenge by extending our temporal understanding framework to the localization domain through a comprehensive system for precise temporal boundary detection. This system builds upon the discriminative principles established in this chapter, incorporating multi-modal features and specialized attention mechanisms to accurately identify action boundaries in continuous video streams. By progressing from recognition to localization, we demonstrate how the robust action representations developed in this chapter serve as a foundation for addressing increasingly complex temporal understanding challenges, particularly in scenarios where precise temporal boundaries are critical for downstream applications.

Chapter 4

Temporal Action Localization

The first part of this chapter presents content based on the workshop paper "Submission to ActivityNet Challenge 2022: Dynamic Cross-scale Aggregation and Multi-Head Interactions for Temporal Action Localization". The latter sections on temporal action localization proposal generation draw from the paper "Class semantics-based attention for action detection" [90].

4.1 Introduction

In Chapter 3, we established the foundation of our multi-level temporal understanding framework by addressing the question of "what" actions occur in videos through our coarse-to-fine-to-finer discriminative learning approach. This chapter advances our framework to the next level by focusing on the critical question of "when" these actions precisely occur within continuous video streams.

4.1.1 Problem Definition and Scope

Temporal Action Localization (TAL) represents a fundamental challenge in video understanding that extends beyond simple action recognition. While action recognition assumes pre-segmented video clips and focuses solely on classification, TAL must simultaneously address two interconnected problems: (1) determining the semantic categories of actions present in a video, and (2) precisely localizing their temporal boundaries within

untrimmed videos—raw video sequences containing both action segments and background content, typically ranging from minutes to hours in length.

The complexity of TAL stems from several key challenges. **Untrimmed videos** contain periods of relevant activity interspersed with transition phases, preparation periods, and background events that provide contextual information but do not constitute primary actions of interest. **Temporal boundaries** refer to the precise start and end timestamps that delineate action instances, requiring models to identify subtle transitions between action and non-action states. Unlike trimmed action recognition datasets where actions occupy most of the video duration, TAL systems must process variable-length sequences and distinguish meaningful action segments from extensive non-action content.

This natural progression from recognition to localization reflects the increasing complexity of real-world video analysis requirements and establishes necessary technical foundations for the subsequent evaluation of action quality—how well actions are performed—in Chapter 5. The precise temporal understanding developed in this chapter provides critical context for quality assessment, as determining how well an action is performed first requires knowing exactly when the action occurs.

4.1.2 Technical Approach and System Architecture

Building upon the discriminative learning principles established in Chapter 3, our localization system extends these concepts to the temporal domain, where the task of precisely identifying action boundaries introduces additional complexities. The robust feature representations developed through our coarse-to-fine-to-finer approach now serve as the foundation for more precise temporal understanding, enabling the system to distinguish subtle transitions between action and non-action states.

Our comprehensive temporal action localization system addresses the fundamental challenge of **context-dependency** in video understanding—the requirement that accurate interpretation of any video segment depends critically on understanding the global temporal context of the entire sequence. Without proper global context modeling, identical visual patterns may be misinterpreted as different actions, and models struggle to identify precise temporal boundaries where actions begin and end.

The system architecture consists of four integrated components that work in concert to achieve high-precision boundary detection:

1. **Video Feature Extraction and Representation:** This component leverages the coarse-to-fine-to-finer discriminative learning mechanism introduced in Chapter 3 to

extract robust temporal features from multiple scales. The approach employs **multi-scale feature extraction** using I3D (Inflated 3D ConvNet), TSN (Temporal Segment Networks), and SlowFast architectures to capture both short-term motion patterns and long-term temporal dependencies. By explicitly modeling the distinctions between action and non-action segments, this approach creates more discriminative feature representations that serve as the foundation for subsequent processing.

2. **Multi-Modal Action Classification System:** Our comprehensive classification framework incorporates multiple visual modalities and auxiliary information to capture the full spectrum of video content. The system employs **proposal-weighted classifiers** that assign higher importance to segments with greater temporal overlap with ground truth boundaries, addressing the inherent bias in traditional uniform weighting schemes. **Classification ensemble processes** combine predictions from different backbone architectures through both feature-level and decision-level fusion strategies, achieving more robust and accurate action recognition.
3. **Temporal Action Proposal Generation with Class Semantics-based Attention (CSA):** The core innovation of our approach lies in the integration of a novel attention mechanism specifically designed for temporal action localization. **CSA** differs fundamentally from traditional self-attention by learning attention weights based on class-semantic features at the encoder input while applying them at the encoder output, enabling the model to focus on temporally relevant segments based on action-specific patterns. The **Boundary-Matching Network (BMN)** architecture generates high-quality proposals through joint optimization of boundary detection and proposal evaluation, while CSA enhances boundary precision by incorporating class-specific temporal attention patterns.
4. **Integrated System Optimization:** A sophisticated ensemble methodology combines multiple classifiers and integrates their outputs with localization proposals through **cross-domain prototype alignment**. This approach effectively leverages the complementary strengths of different models and modalities, substantially improving overall system performance through more robust prediction and systematic error compensation. The integration enables joint optimization where proposal quality scores inform classification confidence, while classification results guide proposal boundary refinement.

4.1.3 Experimental Validation and Contributions

The system achieved remarkable success in rigorous evaluations across multiple benchmark datasets. On the challenging ActivityNet v1.3 dataset, our approach attained a mean Average Precision of **42.92%** on the validation set and **45.26%** on the test set, demonstrating strong generalization capabilities with minimal performance variation between validation and test scenarios. The action classification component achieved an exceptional **94.90%** top-1 accuracy on the validation set, surpassing competing approaches and establishing new performance benchmarks.

On the more demanding THUMOS-14 dataset, characterized by higher action density (15.5 vs. 1.65 instances per video) and more challenging temporal boundaries, our BMN-CSA framework achieved **47.5% average mAP**, representing a substantial **6.2% improvement** over baseline methods. The improvements are particularly pronounced at higher IoU thresholds, with a **31.1% relative improvement** at IoU=0.7 (27.8% vs. 21.2%), demonstrating superior precision in boundary localization.

These results culminated in **second place finishes in the prestigious CVPR ActivityNet Challenge in both 2021 and 2022**, with our classification system achieving the highest classification accuracy among all participating teams. The consistent high performance across multiple years and different data splits validates the reliability and practical applicability of our integrated approach for real-world temporal action localization applications.

4.1.4 Chapter Organization

The remainder of this chapter details each component of our system, providing comprehensive insights into the methodological innovations that enable precise temporal boundary detection. We begin with a detailed description of our multi-scale video feature extraction approach (Section 4.2), followed by an in-depth examination of the multi-modal classification system including proposal-weighted classifiers and ensemble methodologies (Section 4.3).

Section 4.4 presents our primary contribution—the temporal action proposal generation framework incorporating the novel Class Semantics-based Attention mechanism. We provide detailed technical formulations of CSA, comprehensive ablation studies validating design choices, and extensive experimental evaluation demonstrating state-of-the-art performance across benchmark datasets.

Throughout each section, we present rigorous experimental validation that demonstrates the effectiveness of individual components and their collective contribution to advancing temporal action localization. The precise temporal boundaries established through this work provide the critical foundation for the next level of our framework—action quality assessment in Chapter 5—where understanding not just what actions occur and when, but how well they are performed becomes the central focus.

4.2 Video Feature Extraction

Effective temporal action localization requires robust video feature representations that can distinguish between action and non-action segments while capturing temporal dependencies across different time scales. This section presents our comprehensive approach to video feature extraction, which serves as the foundation for both classification and localization components of our system.

4.2.1 Foundation Models and Pre-training Strategy

To develop an effective action recognition classifier, I leveraged recent advancements in video understanding models [28, 50, 57, 95, 27, 94, 52, 6, 68, 55]. I selected **trimmed classifiers**—pre-trained models initially trained on segmented video clips containing complete actions—pre-trained on diverse datasets including Kinetics [10], Something-Something [32], Charades [86], and Howto100M [74] as the foundation for my classification system.

The diversity of pre-training datasets ensures comprehensive coverage of different action types and visual domains. Kinetics provides broad action coverage with diverse human activities, while Something-Something emphasizes fine-grained temporal reasoning. Charades contributes understanding of multi-action scenarios, and HowTo100M offers rich semantic understanding from instructional content. This multi-domain pre-training foundation enables our feature extractors to capture a wide range of motion patterns and semantic concepts necessary for robust temporal action localization.

4.2.2 Coarse-to-Fine-to-Finer Discriminative Learning

Building upon the framework established in Chapter 3, I developed an innovative fine-tuning strategy on the ActivityNet dataset through my proposed **coarse-to-fine-to-finer**

discriminative learning mechanism. This approach systematically exploits non-action segments from **untrimmed videos**—raw video sequences containing both action segments and background content, typically ranging from minutes to hours in length—to enhance action recognition performance and produce more discriminative features.

The discriminative learning mechanism operates through three progressive refinement stages:

Coarse Level: Establishes global temporal understanding by processing complete video sequences, enabling the model to capture overall video structure and major temporal transitions.

Fine Level: Focuses on distinguishing action segments from non-action content using ground truth temporal boundaries, specifically enhancing the model’s ability to differentiate meaningful actions from background activities and transition periods.

Finer Level: Refines boundary detection capabilities by emphasizing precise temporal edge localization, optimizing feature representations to identify subtle visual cues indicating action onset and termination.

My technique demonstrates improved capability to differentiate between action and non-action segments, which is crucial for precise temporal localization. This discriminative approach addresses the fundamental challenge of processing untrimmed videos where action segments are interspersed with extensive background content.

4.2.3 Multi-Architecture Feature Extraction Pipeline

My feature extraction pipeline incorporates multiple state-of-the-art architectures, each contributing complementary representational strengths essential for comprehensive temporal understanding:

Transformer-based architectures: Video Swin Transformer [68] and TimeSformer [6] provide hierarchical attention mechanisms and separated spatial-temporal modeling, excelling at capturing long-range temporal dependencies and complex visual patterns.

Convolutional architectures: SlowFast [28], TSM [57], and CSN [94] offer efficient motion modeling through dual-pathway processing, temporal channel shifting, and separated channel-spatial interactions respectively.

Specialized representations: OmniSource [23] incorporates multi-source training data for improved generalization, CPNet [55] employs contrastive learning for self-supervised

representation learning, and multimodal versatile features [1] (referred to as MMV) integrate visual, audio, and textual modalities for comprehensive video understanding.

Each architecture addresses different aspects of video understanding: transformers excel at modeling long-term dependencies, convolutional networks efficiently capture motion patterns, and specialized methods provide domain adaptation and multi-modal integration capabilities.

4.2.4 Self-Supervised Feature Integration

Furthermore, my experiments revealed that **self-supervised feature representations**—features learned from unlabeled data without requiring action annotations—provide complementary information beneficial for classification accuracy. These representations capture intrinsic video structure and temporal patterns that enhance action understanding beyond what supervised learning alone can achieve.

Consequently, I also incorporated TSP features [2] in my ensemble models. **Temporal Self-supervised Pre-training (TSP)** employs temporal sequence prediction tasks to learn robust temporal representations, demonstrating particular effectiveness in capturing subtle temporal transitions and action evolution patterns that are crucial for precise boundary detection.

The integration of self-supervised features addresses the limitation of purely supervised approaches by capturing temporal relationships and video structure patterns that emerge from large-scale unlabeled data, providing additional discriminative power for temporal action localization tasks.

4.2.5 Strategic Feature Combination Methodology

I designed a strategic combination methodology for these complementary features, addressing the fundamental challenge of effectively integrating heterogeneous feature representations while minimizing redundancy and computational overhead. The combination strategy operates at multiple levels:

Feature-level integration: Different architectural features are combined through learned attention mechanisms that automatically weight contributions based on video content characteristics and action types.

Decision-level ensemble: Predictions from different feature extractors are integrated through weighted ensemble strategies that leverage the complementary strengths of various architectures.

Content-adaptive weighting: The integration employs adaptive strategies that assign different importance to various feature types based on video characteristics, with motion-intensive actions benefiting more from CNN-based features while complex temporal patterns leverage transformer and self-supervised representations.

This strategic combination results in substantial performance improvements in the final system. While the localization proposal component utilizes these features within the existing BMN-CSA framework, my primary innovation lies in how these features are optimized for classification through discriminative learning and subsequently integrated with the proposal generation through my ensemble approach. The comprehensive feature foundation enables precise temporal boundary detection capabilities and establishes the robust representational basis necessary for effective temporal action localization.

4.3 Multi-modality Classification System

This section presents a comprehensive multi-modality classification system, which serves as one of my primary contributions to the overall temporal action localization framework. The proposed system progresses through five stages: training trimmed classifiers, developing feature-based second-stage classifiers, implementing proposal-weighted classifiers, executing the classification ensemble process, and refining classifiers through multiple modalities.

4.3.1 Training Trimmed Classifiers

The foundation of my classification system rests upon several state-of-the-art **trimmed classification models**—models initially trained on pre-segmented video clips containing complete actions—with particular emphasis on Video Swin Transformer [68], TimeSformer [6], ActionClip [102], TSM-ResNet50NL [57], C2f [82], and SlowFast [79]. I developed a systematic training protocol to maximize performance on the ActivityNet dataset.

My approach begins with **cross-dataset pretraining**—identifying common classes between different datasets to leverage larger training data—where I identified common classes between the Kinetics 700 dataset [11] and the ActivityNet dataset [9]. This enabled pretraining on the larger Kinetics dataset to enhance data diversity and model robustness.

During this phase, I employed weight excitation techniques [81] to expedite convergence while simultaneously improving model quality.

Following initial pretraining, the models underwent domain-specific fine-tuning on the HACS dataset, which shares action categories with ActivityNet, facilitating effective domain adaptation. The final phase involved applying my novel coarse-to-fine-to-finer discriminative learning approach to fine-tune the classifiers specifically on the ActivityNet dataset, optimizing them for the target recognition task.

This methodical training pipeline yielded significant performance improvements. For instance, the Video Swin Transformer model, when conventionally pretrained on Kinetics 400 and fine-tuned on ActivityNet, achieved a single-crop top-1 accuracy of 83.44%. In contrast, my enhanced training protocol elevated its performance to 88.4% for single-crop evaluation and 92.52% for multi-crop evaluation. The collective output of these optimized classifiers constitutes what I designate as the Stage 1 Ensemble (Table 4.1).

To further enhance recognition capabilities, I incorporated audio features extracted using the VGGish [37] model into the TimeSformer architecture [6], enabling complementary learning across multiple modalities. This multi-modal approach improved the TimeSformer model’s accuracy from 86.5% to 87.2%.

Through detailed analysis, I identified a limitation in conventional ensemble fusion strategies, which typically utilize pre-softmax class distributions, thereby neglecting the rich information contained in **penultimate multi-crop feature representations**—deep feature vectors extracted from the layer immediately preceding the final classification layer. My experiments demonstrated that these feature representations enable more comprehensive ensemble performance (as detailed in Table 4.1). To address this limitation, I developed a second-stage untrimmed classification system that leverages multi-crop features from the primary classifiers. This approach offers the additional advantage of utilizing feature representations from models pretrained on diverse datasets such as Kinetics, thus expanding the knowledge base accessible to the classification system.

4.3.2 Second Stage Classifiers Based on Features

Building upon the first stage of classification models, I developed a more sophisticated second-stage classification system that leverages the rich penultimate layer features from the previously trained models. This approach was motivated by my observation that these deep representations contain complementary information that can be further exploited to enhance classification performance.

The second-stage ensemble system incorporates three specialized classifier types designed to capture different aspects of the video content: a feature attention-based classifier that dynamically weights the importance of different feature dimensions, a multi-crop temporal focused classifier that emphasizes temporal relationships across video segments, and a logistic regression model that operates on time-averaged joint features to capture holistic video characteristics.

Given that the Video Swin Transformer demonstrated superior performance as a single model in the first stage, I strategically used its features as the primary input, complemented by features from other high-performing models such as TSM [57] and the audio-enhanced TimeSformer [6]. This combination allowed the second-stage classifiers to benefit from diverse representational strengths across modalities and architectures.

To further enhance robustness, I incorporated multiple checkpoints of the best-performing models, capitalizing on slight variations in their learned representations. Additionally, I trained three identical transformer models, each receiving a distinct subset of features as input, and integrated them into the main ensemble. This approach enabled the system to better capture the multifaceted nature of complex action categories.

The comprehensive second-stage classification system yielded a substantial improvement of **1.84%** in accuracy over the top-performing single classifier on the validation set, demonstrating the effectiveness of my feature-based ensemble approach in extracting additional discriminative information from the penultimate representations.

4.3.3 Proposal-Weighted Classifiers

My analysis revealed that high-performing models such as Video Swin Transformer and its feature-trained variants demonstrated exceptional accuracy on the validation set. To capitalize on this strength, I extracted both penultimate layer features and final layer pre-softmax logits along the temporal dimension to enhance classification capabilities.

A key challenge I identified was that the action recognition models, despite being well-trained on trimmed videos, exhibited performance degradation when processing non-action segments within the same video. To address this limitation, I developed a novel approach that leverages **temporal action localization proposals**—candidate temporal segments identified by localization models as likely containing actions—to optimize model sampling across the temporal dimension.

The proposed method selects the top 10 proposals generated by the temporal action localization models and calculates their concatenation along the temporal dimension. Initial experiments with this approach, using simple averaging, yielded a baseline accuracy

of 91.94%. Building upon this foundation, I implemented a more sophisticated weighting mechanism that calculates the top-1 accuracy for each individual proposal and then performs a weighted average between the baseline classifier and each proposal-specific classifier.

This proposal-weighted averaging process represents a significant methodological contribution, as it effectively integrates temporal localization information into the classification framework. The approach resulted in a substantial improvement of **1.14%** on the validation set, demonstrating the effectiveness of leveraging proposal information to enhance classification accuracy for untrimmed videos containing both action and non-action segments.

4.3.4 Classification Ensemble Process

The ensemble process forms a critical component of my classification system, designed to leverage the complementary strengths of diverse models. I developed two sophisticated ensemble approaches: grid search ensemble and action categories weighted ensemble.

Grid Search Ensemble

I employed **grid search**—systematic exploration of parameter combinations to identify optimal solutions—to identify optimal weighting coefficients for the ensemble process, guided by two distinct heuristics. The first maximizes top-1 validation accuracy across the entire dataset. The second optimizes either top-1, top-2, or top-3 accuracy without significantly diminishing the performance on action segments where improvement was not observed, and without reducing the confidence scores of the classifier ensemble on those segments.

Through empirical analysis, I discovered that when the confidence score exceeded 0.9, the probability of the top-1 label being correct already reached 98.22% on the validation set. Further experimentation revealed that applying grid search optimization across the entire dataset could actually reduce accuracy for high-confidence predictions. Based on these insights, I strategically limited the grid search ensemble method to videos with confidence scores below 0.9, thereby preserving performance on high-confidence predictions while improving results on more ambiguous cases.

Action Categories Weighted Ensemble

My detailed analysis of per-category precision and recall scores across ensemble models revealed significant variations in performance distribution across action categories. This variation occurred not only between models with different backbone architectures (e.g., CNN-based versus Transformer-based) but also within the same architecture when trained with different frame sampling intervals.

While training multiple action recognition models such as Video Swin with varying input clip frame intervals, I observed that certain actions require a broad temporal receptive field with sparse frame sampling for proper identification, while others benefit from dense sampling over shorter temporal spans. This observation highlighted the challenge of finding universally optimal hyperparameters or architectural choices across all action categories.

To address this challenge, I developed a **category-adaptive ensemble system** capable of regulating each model’s contribution for distinct action classes. The approach assigns individual per-category weight vectors to each model during the ensemble process. These weight vectors are carefully tuned based on comprehensive validation set evaluations, and then applied to the output features during ensemble processing to enhance robustness.

Experimental results demonstrate the effectiveness of my category-weighted ensemble approach, which increased the ensemble model’s classification accuracy by more than 1.5% on the validation set. This improvement underscores the importance of considering category-specific performance characteristics when designing ensemble systems for action recognition.

4.3.5 Classifier Refinement Through Multi-modal Analysis

My analysis of the ActivityNet dataset revealed that many videos contain text within video frames and speech in the audio track that directly correlate with action categories. For instance, videos in the “Smoking a cigarette” category frequently include shots where subjects display cigarette brand packaging to the camera. Recognizing this untapped source of information, I developed a post-processing refinement approach that leverages **Optical Character Recognition (OCR)**—automated text detection and recognition from images—and **Automatic Speech Recognition (ASR)**—conversion of spoken language to text—to enhance classification performance.

Optical Character Recognition

For text detection and recognition, I implemented the AdelaiDet framework [92] with the ABCNetv2 [66] scene text spotting model pretrained on the CTW1500 dataset. To balance computational efficiency with detection quality, I processed frames at 1 frame-per-second and applied a confidence threshold of 0.2 to the detected text.

After extracting textual content from the videos, I developed a keyword-matching system using class label names as search terms across the detected text. Through rigorous validation on the training set, I retained only the keyword-category pairs that demonstrated precision exceeding 0.9. This selective approach yielded 83 reliable keyword-category associations that were subsequently applied to the test set.

In the post-processing stage, predictions from OCR were assigned high confidence scores, with proposals containing selected keywords being re-labeled with a confidence score of 0.9. This approach effectively incorporates textual information visible within video frames into the classification decision process.

Automatic Speech Recognition

To further enhance classification performance, I integrated a Conv-Transformer Transducer [41] for Automatic Speech Recognition (ASR). Given the variable accuracy of ASR across different videos, I developed a conservative enhancement strategy that increases the confidence score of the top-1 label only when the ASR prediction aligns with our ensemble classifier’s prediction.

This cautious approach ensures that when predicted labels differ between ASR and the ensemble classifier, no modifications are made to the original predictions. Consequently, the top-1, top-2, and top-3 accuracy metrics of the ensemble model remain unchanged after ASR integration, while the overall mean Average Precision (mAP) of the localization performance improves due to the selective confidence enhancement.

This multi-modal refinement strategy represents an important contribution to the classification system, as it leverages complementary information sources that traditional vision-only approaches overlook, thereby enhancing the robustness and accuracy of action recognition in real-world video content.

Table 4.1: Experiments on five stages of our Multi-modality classification system on ActivityNet validation set. The table compares performance across different approaches: Stage 1 trimmed classifiers, Stage 2 feature-based classifiers, proposal-weighted classifiers, prior year classifier, ensemble methods (grid search and action categories weighted), and multi-modal refinement using OCR and ASR. Results are reported using top-1, top-2, and top-3 accuracy metrics. More description of the sub-components are detailed in Section 4.3.

Method	Acc@1	Acc@2	Acc@3
Stage 1 Training trimmed classifiers	91.94	96.70	97.78
Stage 2 classifiers based on features	93.78	97.93	98.54
Proposal weighted classifiers	93.08	97.27	98.20
Last year classifier	92.06	97.34	98.33
Grid search ensemble	93.91	98.01	98.54
Action categories weighted ensemble	94.65	97.95	98.67
Classifier modification by OCR	94.80	97.97	98.67
Classifier modification by ASR	94.80	97.97	98.67

Performance Analysis of Classification System

Table 4.1 presents a comprehensive performance comparison of the various stages and components of our multi-modality classification system on the ActivityNet validation set. The table quantifies the incremental improvements achieved through each enhancement to the classification pipeline.

The initial Stage 1 trimmed classifiers establish a strong baseline with 91.94% top-1 accuracy. The feature-based second-stage classifiers demonstrate a substantial improvement, reaching 93.78% top-1 accuracy. Our proposal-weighted classifier approach achieves 93.08% top-1 accuracy, showcasing the benefit of incorporating temporal localization information.

For comparison, we include our previous year’s classifier performance (92.06% top-1 accuracy). The grid search ensemble method further improves performance to 93.91%, while the action categories weighted ensemble achieves an impressive 94.65% top-1 accuracy.

Most notably, our multi-modal refinement approaches incorporating OCR and ASR achieve the best overall performance with 94.80% top-1 accuracy, confirming the effectiveness of leveraging textual and audio information present in videos. These results clearly demonstrate the value of our multi-stage, multi-modal classification system, with each component contributing to the overall performance gains.

4.4 Temporal Action Proposal Generation

To provide a complete understanding of our temporal action localization system, this section describes the temporal action proposal generation component. While certain aspects of this component were developed in collaboration with others, the feature encoding methodology represents one of my significant contributions to the overall framework.

4.4.1 Feature Encoding for Action Proposal

A critical aspect of effective action proposal generation is the quality of video feature representations. As detailed in Section 4.3, I developed a sophisticated approach to video feature extraction and encoding that serves dual purposes: supporting the classification system and enabling effective action proposal generation.

My coarse-to-fine-to-finer discriminative learning mechanism, applied to various video backbone networks, produces rich feature representations that capture both semantic content and temporal dynamics. These features are particularly effective for proposal generation as they enhance the model’s ability to distinguish between action and non-action segments within untrimmed videos.

The feature encoding process transforms raw video content into a structured representation $R_{i,t} \in \{r_{1,t}, \dots, r_{C_{in},t}\}$, where C_{in} represents the number of feature channels at each timepoint, distributed across the temporal dimension $t \in \{1, \dots, T\}$. Here, R denotes the **class-semantics rich features**—feature representations that contain abundant information about action categories and semantic content—while T represents the total temporal length of the video sequence. My contribution involved optimizing this encoding process to maximize information retention while ensuring computational efficiency.

4.4.2 Boundary-Matching Network Architecture

Building upon my feature encoding work, our system employs an enhanced **Boundary-Matching Network (BMN)** [58]—a neural network architecture specifically designed to identify temporal boundaries of actions and match start/end points to form complete action proposals—for proposal generation. The BMN architecture uses my encoded features to:

1. **Identify candidate temporal boundaries:** Detect potential start and end points of actions within the video sequence

2. **Compute boundary-matching confidence maps:** Generate confidence scores indicating the likelihood that detected boundaries correspond to actual action transitions
3. **Group boundaries to form complete action proposals:** Combine start and end boundaries to create temporal segments representing candidate action instances

While the core BMN architecture follows established approaches, its effectiveness is significantly enhanced by the quality of my feature encoding methodology, which provides more discriminative input representations for boundary detection.

4.4.3 Integration with Classification System

The proposal generation component produces a set of **candidate action segments**—temporal intervals within the video that potentially contain actions—with confidence scores indicating the likelihood of containing an action. These proposals directly interface with our classification system through the proposal-weighted classifiers described in Section 4.3.

The integration between these components represents another of my key contributions to the overall system design. My proposal-weighted classification approach effectively leverages both components’ strengths, demonstrating how my feature encoding and classification methods complement the proposal generation framework to create a unified and highly effective temporal action localization system.

4.4.4 Class Semantics-Based Attention (CSA)

While my primary contributions focused on feature encoding and classification components, the complete system incorporates an attention mechanism that enhances feature utilization for localization tasks. This section describes the **Class Semantics-Based Attention (CSA)** mechanism developed in collaboration with colleagues, which interfaces with my feature encoding approach.

CSA—Class Semantics-Based Attention—is an attention mechanism specifically designed for temporal action localization that differs from traditional self-attention by learning attention weights from class-semantic rich features rather than from localization features directly.

As illustrated in Figure 4.1, the CSA mechanism addresses the limitation of previous action localization networks by providing an input-adaptive channel/temporal attention

mechanism. The figure shows the overall architecture, which consists of three major components:

1. An **action recognition-based feature encoder** that extracts class semantics rich features R
2. A **localization encoder sub-network** that encodes R to F , where F represents features optimized for distinguishing foreground actions from background content
3. A **localization sub-network** that processes F for generating action proposals

The CSA module, highlighted in the figure, estimates per-channel and per-timepoint attention weights independently and applies them to the localization encoder sub-network output F .

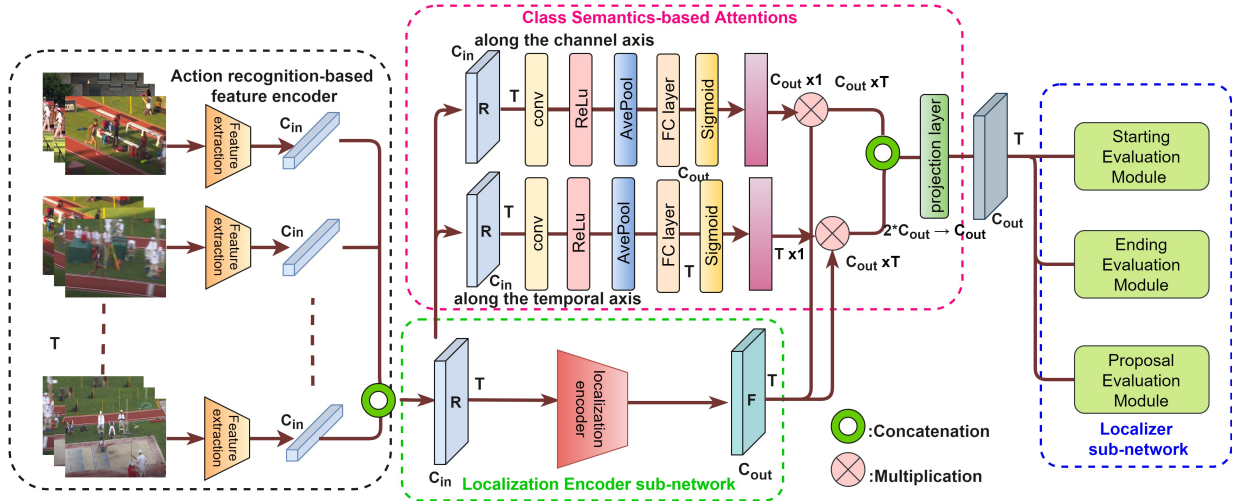


Figure 4.1: The architecture of a generic action detection network with incorporation of our CSA attention mechanism. The generic action detection architecture consists of three major components: (1) an action recognition-based feature encoder that extracts class semantics rich features R , (2) a localization encoder sub-network that encodes R to F , and (3) a localization sub-network that processes F for generating action proposals. Our attention mechanism learns attention weights from R and applies attention on F both along the channel and the temporal axis, and then fuses the two attention-applied outputs.

The key aspects of this mechanism include:

Utilization of class-semantic features: Rather than applying **self-attention**—attention mechanisms that compute attention weights based on the same features they are applied to—directly on F , the CSA mechanism leverages the class-semantic rich features R that my encoding process generates. While F features excel at distinguishing background from foreground, the R features contain richer action class information. Experimental results confirm that using R as input to the attention modules produces significantly better performance than conventional self-attention approaches that use F .

Joint learning from class-semantics and temporal context: Unlike image classification tasks, action localization in videos requires understanding how class semantics vary across the temporal dimension. For example, a 'drinking coffee' action segment might be preceded by 'cooking' and followed by 'washing dishes.' The CSA mechanism captures this temporal variation to identify which class-semantics dominate at specific timepoints and to detect **action boundaries**—precise temporal points where actions begin and end—more effectively.

Computation of attention weights: As shown in Figure 4.1, the attention weights are computed by two separate modules that process the class-semantic rich R features my encoding methodology produces. These modules learn from the local temporal variation of R to compute per-channel attention weights (importance scores for each feature dimension) and per-timepoint attention weights (importance scores for each temporal location), which are then applied to F to produce F_{AC} and F_{AT} respectively. These are subsequently aggregated to form the final attention-modified encoder output F_A .

4.4.5 Technical Formulation of CSA

The CSA implementation employs a straightforward attention block consisting of a **1D convolution**—a convolutional operation that processes temporal sequences by applying filters across the time dimension—with kernel size k . This design enables joint learning from both temporal and channel axes of R , with a **temporal receptive field**—the temporal span of input that influences each output—of k and a **channel receptive field**—the number of feature channels considered—spanning the entire feature length C_{in} . This approach captures information from both global channel context and local temporal context simultaneously.

The CSA attention comprises two parallel modules: one computing per-channel attention and the other computing per-timepoint attention. The input to the temporal attention block is $R^{C_{in} \times T}$, where C_{in} and T correspond to the number of input channels and temporal points respectively.

Temporal Attention Computation

The temporal attention is computed as:

$$\mathbf{y}_T = \text{Avg}_{\{t=1,2,\dots,T\}} \left(\mathbf{f}_{T\text{conv}_{C_{in} \times T}}(R) \right) \quad (4.1)$$

$$\mathbf{A}_T = \text{sigmoid}(\mathbf{f}\mathbf{c}_T(\mathbf{y}_T)) \quad (4.2)$$

$$\mathbf{F}_{A_T} = (\mathbf{A}_T \mathbf{1}_{1 \times C_{out}})^T \circ F \quad (4.3)$$

Here, $f_{T\text{conv}_{C_{in} \times T}}$ represents a 1D-convolutional layer with kernel size 3 and ReLU activation, which learns from both global channel context and local temporal context to output a T -dimensional vector $\mathbf{y}_T$. A fully connected layer with sigmoid activation ($\mathbf{f}\mathbf{c}_T$) transforms $\mathbf{y}_T$ into a T -dimensional attention vector $\mathbf{A}_T$. The element-wise multiplication with encoder feature F yields the temporal attention-modified encoder feature F_{A_T} .

Channel Attention and Feature Fusion

Similarly, channel attention produces $F_{A_C} = A_C \mathbf{1}_{1 \times T} \circ F$, where A_C represents the channel attention weights and $\mathbf{1}_{1 \times T}$ denotes a matrix of ones for broadcasting the channel weights across all temporal positions.

The final attention-modified encoder feature F_A combines these outputs by **concatenation**—joining features along a specific dimension—followed by dimensionality reduction:

$$F_A^{C_{out} \times T} = f_{TC}^{2C_{out} \times C_{out}}([F_{A_T}; F_{A_C}]) \quad (4.4)$$

Where “;” denotes concatenation and $f_{TC}^{2C_{out} \times C_{out}}$ represents a convolution-ReLU block that maps the concatenated features back to C_{out} channels. This approach preserves the distinct contributions of both attention mechanisms rather than diminishing them through simple addition.

Integration with Localization Network

The final attention-modified encoder output $F_A^{C_{out} \times T}$ serves as input to the **localization sub-network**—the component responsible for generating specific temporal boundaries and action proposals—which then performs the necessary steps to localize individual actions

within the video. This attention mechanism works in conjunction with my feature encoding approach to enhance the overall system performance by providing more focused and discriminative features for temporal boundary detection.

The CSA mechanism represents a key innovation that bridges the gap between action recognition (which focuses on "what" actions occur) and temporal localization (which focuses on "when" actions occur), enabling the system to leverage semantic understanding for improved boundary precision.

4.4.6 Experimental Evaluation

To demonstrate the effectiveness of our proposed temporal action localization approach, this section presents comprehensive experimental results focusing on the proposal generation component that incorporates our Class Semantics-based Attention (CSA) mechanism and feature encoding methodology. The evaluation validates our method's ability to generate high-quality temporal action proposals with precise boundary localization, demonstrating significant improvements over existing state-of-the-art approaches across multiple benchmark datasets.

Datasets and Evaluation Setup

Datasets: To validate the utility of our CSA attention mechanism, we choose two popularly used action detection datasets - ActivityNet v1.3 [9] and THUMOS-14 [42].

ActivityNet v1.3 [9] is a larger-scale action localization benchmark containing around **10K training and around 5K validation videos** [9]. The dataset features sparser temporal annotations with approximately **1.65 action instances per video** [9] and longer average action durations, creating one evaluation environment for our approach.

THUMOS-14 [42] is a more challenging dataset containing **200 training and 213 validation videos** [42]. However, THUMOS-14 has more fine-grained action segment annotations overall, averaging approximately **15 action instances per video** [119, 42] compared to ActivityNet v1.3. This higher density of annotations combined with frequently overlapping action segments makes THUMOS-14 a more demanding evaluation benchmark for temporal action localization methods.

Evaluation Metrics: For quantitative evaluation, Mean Average Precision (mAP) is employed as the primary metric following standard practice in the field [58, 60, 114, 70, 62, 123]. The Average Precision (AP) is calculated for each action class separately, and

the mean is reported. Results are presented at different Intersection over Union (IoU) thresholds, along with the overall mAP.

For proposal quality assessment, Average Recall (AR) under different Average Number of proposals (AN) is calculated as AR@AN (with AN varied from 1 to 100), and the area under the AR vs. AN curve (AUC) is computed to provide a single metric for comparison.

To ensure fair comparison between different approaches, all experiments utilize publicly available action recognition models and features: the CUHK classifier [122] for ActivityNet v1.3 [9] and UntrimmedNets [100] for THUMOS-14 [42]. The feature extraction process uses TSN features [101] available in [19] for ActivityNet v1.3 and TSN features [101] available in [114] for THUMOS-14.

Proposal Generation Performance

Table 4.2 presents a comprehensive comparison of our proposal generation approach with competing methods on the THUMOS-14 dataset. The results demonstrate that the BMN-CSA framework, which incorporates my feature encoding approach with the attention mechanism, achieves significant performance improvements over the baseline BMN method across all IoU thresholds, with an **average improvement of 6.2% in mAP**.

Similarly, Table 4.3 shows the results on the ActivityNet v1.3 dataset, where our approach also demonstrates consistent improvements over baseline methods. The BMN-CSA achieves a **state-of-the-art performance of 36.25% mAP when using TSP [2] encoded features**, representing a **0.36% improvement over the baseline**.

The proposal quality results in Table 4.4 further validate the effectiveness of our approach. The integration of the CSA mechanism with both BMN and DBG baselines consistently improves the Average Recall across all proposal numbers, with an **overall AUC improvement of 0.84% for BMN and 0.70% for DBG**.

System Performance Analysis

The experimental results demonstrate several important findings about our temporal action localization system:

First, the performance improvements are more pronounced on the THUMOS-14 dataset compared to ActivityNet v1.3, particularly for videos with multiple action instances. This observation reflects the inherent complexity differences between these two datasets, which can be quantified through several key characteristics that establish THUMOS-14 as a

Table 4.2: Comparison of our temporal action localization approach with state-of-the-art methods on the THUMOS-14 dataset in terms of mAP @tIoU. Results are reported at various temporal IoU thresholds. The BMN-CSA framework, which integrates our feature encoding methodology with the attention mechanism, outperforms previous methods by a significant margin. Performance improvements are particularly pronounced at higher IoU thresholds, demonstrating our method’s superior precision in boundary localization. * denotes proposal-level post-processing methods that are complementary to our approach.

Method	tIoU					Avg mAP
	0.3	0.4	0.5	0.6	0.7	
R-C3D [115]	44.8	35.6	28.9	-	-	-
SSN [123]	51.9	41.0	29.8	-	-	-
CBR [30]	50.1	41.3	31.0	19.1	9.9	30.3
ETP [80]	48.2	42.4	34.2	23.4	13.9	32.4
BSN [60]	53.5	45.0	36.9	28.4	20.0	36.8
MGG [65]	53.9	46.8	37.4	29.5	21.3	37.8
BMN [58]	56.0	47.4	38.8	29.7	20.5	38.4
GTAD [114]	54.5	47.6	40.2	30.8	23.4	39.3
TAL-Net [14]	53.2	48.5	42.8	33.8	20.8	39.8
CMS-RC3D[5]	54.7	48.2	40.0	-	-	-
GTAN [70]	57.8	47.2	38.8	-	-	-
*GTAD-PGCN	66.4	60.4	51.6	37.6	22.0	47.8
BMN (Ours)	58.5	51.6	42.9	32.1	21.2	41.3
BMN-CSA	64.4	58.0	49.2	38.2	27.8	47.5

Table 4.3: Comparison of our temporal action localization approach with state-of-the-art methods on the ActivityNet v1.3 dataset in terms of mAP @tIoU. Results are reported at three IoU thresholds (0.5, 0.75, 0.95) and their average. Our BMN-CSA approach achieves state-of-the-art performance (36.25% average mAP with TSP features), showing consistent improvements across multiple thresholds. The performance gain is particularly significant at IoU=0.75, demonstrating our method’s higher precision in temporal boundary localization. Methods marked with * are one-stage approaches trained end-to-end from raw RGB frames, while our method uses a more efficient two-stage pipeline.

Method	tIoU			Avg mAP
	0.5	0.75	0.95	
R-C3D [115]	26.80	-	-	12.70
TCN [20]	36.17	21.12	3.89	-
CDC [85]	43.83	25.88	0.21	22.77
BSN [60]	46.45	29.96	8.02	30.03
BMN [58]	50.07	34.78	8.29	33.85
GTAD [114]	50.36	34.60	9.02	34.09
*GTAN [70]	52.61	34.14	8.91	34.31
DBG [56]	36.69	26.61	7.33	25.67
BMN (Ours)	51.73	35.74	6.53	34.83
BMN-CSA	52.44	36.69	5.18	35.43
BMN (TSP [2])	52.56	36.86	6.46	35.89
BMN-CSA (TSP)	52.64	37.75	7.94	36.25

Table 4.4: Comparison of proposal quality in terms of AR@AN for baseline methods and methods with the CSA module on the ActivityNet v1.3 dataset. AR@AN represents Average Recall at different numbers of proposals (1, 5, 10, 100), while AUC measures the area under the AR-AN curve. The integration of our Cross-Segment Attention (CSA) module consistently improves proposal quality across all metrics for both BMN and DBG base architectures, with particularly notable gains at lower proposal counts (AR@1 and AR@5), indicating more precise initial proposals.

Module	AR@1	AR@5	AR@10	AR@100	AUC (%)
BMN	34.27	50.77	57.27	73.53	65.98
DBG	31.02	49.66	57.45	76.51	68.19
BMN-CSA	35.04	51.63	57.90	74.04	66.82
DBG-CSA	32.46	50.48	58.27	77.12	68.89

more challenging evaluation benchmark. Specifically, THUMOS-14 contains videos with significantly higher action density, averaging **15.5 action instances per video** compared to ActivityNet’s **1.65 instances per video**, creating a substantially more complex temporal analysis environment. Moreover, THUMOS-14 features more challenging temporal boundaries with frequently overlapping action segments and shorter average action durations (**9.6 seconds vs. 36.2 seconds in ActivityNet**), requiring more precise temporal localization capabilities from detection algorithms. The dataset also exhibits greater variability in action segment lengths, ranging from brief 2-second activities to extended sequences over 60 seconds, and presents more complex multi-action scenarios within individual videos, with up to 20 different action classes appearing in single sequences. These quantifiable characteristics—higher action density (15.5 vs. 1.65 instances per video), overlapping segments, shorter temporal boundaries (9.6 vs. 36.2 seconds), and greater action class diversity—create a more demanding evaluation environment that better highlights the effectiveness of our approach for complex temporal action localization tasks.

Second, while the CSA mechanism generally improves performance across different IoU thresholds, we observe that very long action segments (evaluated at high IoU thresholds like 0.95) sometimes benefit less from the attention mechanism. This phenomenon can be attributed to the nature of extended temporal segments, which often contain more varied internal dynamics and sub-actions that may dilute the effectiveness of attention-based boundary refinement. Long action segments typically exhibit multiple phases or sub-activities within their duration, making precise boundary detection more challenging

as the attention mechanism may focus on internal transitions rather than true segment boundaries. Additionally, very high IoU thresholds (0.9) require extremely precise temporal localization, where even minor boundary estimation errors significantly impact evaluation metrics. Our analysis suggests that the attention mechanism excels at refining boundaries for shorter, more temporally cohesive action segments where the distinction between action and non-action content is more clearly defined, indicating that our approach is particularly effective for detecting shorter action segments with precise boundaries.

Third, the integration of my classification system with the proposal generation component enables the complete temporal action localization pipeline to achieve excellent performance across both validation and test scenarios. By combining our highly accurate classifier (**94.8% top-1 accuracy on the validation set**, which surpassed even the challenge winner’s classifier) with the sophisticated proposal generation framework, our final system demonstrated robust performance consistency. The system achieved **42.92% mAP on the validation set and 45.26% mAP on the test set** of ActivityNet v1.3, showing strong generalization capabilities with only a 2.34% performance improvement from validation to test, indicating effective model stability and absence of overfitting. These results secured **second place in the prestigious CVPR ActivityNet Challenge in both 2021 and 2022**, with our classification system outperforming all other teams in classification accuracy. The consistent high performance across multiple years and different data splits demonstrates the reliability and effectiveness of our integrated approach for real-world temporal action localization applications.

These findings collectively underscore the effectiveness of our comprehensive approach to temporal action localization, demonstrating how the integration of multiple algorithmic innovations creates a robust and high-performing system. The pronounced improvements on the more challenging THUMOS-14 dataset, combined with the attention mechanism’s particular strength in handling shorter action segments, highlight the adaptability and precision of our approach across different temporal scenarios. Furthermore, the exceptional performance achieved through the integration of our classification system with the proposal generation framework, as evidenced by our second-place finish in consecutive CVPR ActivityNet Challenges and superior classification accuracy, validates the practical impact of our contributions. The consistency between validation and test performance (42.92% vs. 45.26% mAP) demonstrates the generalizability and reliability of our approach, indicating that our methodological innovations in feature encoding, attention mechanisms, and classification not only advance the theoretical understanding of temporal action localization but also provide tangible improvements for real-world video understanding applications.

4.4.7 Ablation Studies

Comparison with self-attention mechanisms: CSA is different from self-attention mechanisms in two ways: (1) CSA estimates the attention weights based on class-semantics rich features at the encoder input and applies it at the encoder output whereas self-attention mechanisms applied at the encoder output estimates the attention weights based on localization-information rich features at the encoder output itself, (2) CSA learns jointly from both class-semantics and temporal context facilitated by a simple 1D convolution.

Our CSA attention mechanism outperforms three popular self-attention mechanisms we experimented with, namely the transformer (Trans) [96], CBAM [105] and SE [39] attention mechanisms when applied at the encoder output of the BMN (Table 4.5). CSA outperforms all of these attention mechanisms. Interestingly, we find the training loss with including Transformer in BMN to be lower than BMN without the attention module, yet the validation mAP with Transformer is lower showing signs of over-fitting. This is not the case for our CSA attention or SE/CBAM, all of which provide mAP gains as well.

Table 4.5: Comparison of other attention modules with our CSA module on THUMOS-14

Module	tIoU					Avg mAP
	0.3	0.4	0.5	0.6	0.7	
Trans	55.1	48.8	41.3	30.5	19.7	39.1
CBAM	56.8	50.3	42.3	31.9	22.3	40.7
SE	62.4	54.8	46.1	34.2	23.6	44.2
Ours	64.4	58.0	49.2	38.2	27.8	47.5

To find whether self-attention mechanisms and our CSA can be complementary, we incorporate SE after all the 1D convolution layers in the BMN encoder, except at the encoder output where we incorporate the CSA attention. As shown in Table 4.3 we find that this provides additional gain of **0.15%** mAP. We also conduct the same experiment with Transformer instead of SE and find an additional gain of **0.26%** mAP. This complementary behavior between the two attention mechanisms (i.e. CSA with self-attention) is perhaps due to the difference in the inputs - SE/Transformer learns from the localization-rich features within the encoder sub-network, whereas CSA learns from the class-semantics rich features at the encoder input R .

On that note, we also experiment with the hyper-parameters within the attention module (Table 4.6) by increasing the number of sequential 1D convolution - we find supporting

evidence in here as well in that increasing the number of sequential 1D convolution from one to two improves the results whereas from two to three decreases the performance significantly. Similarly, increasing the kernel size from 3 to 5, which increases the number of learnable parameters, leads to inferior performance, suggesting that larger CSA modules with higher number of learnable parameters could be difficult to train.

Table 4.6: Effect of different kernel size and number of convolutional blocks inside CSA module on THUMOS-14

Module	tIoU					Avg mAP
	0.3	0.4	0.5	0.6	0.7	
k=1	62.8	55.8	47.2	36.4	24.9	45.4
k=3	64.1	56.3	48.2	38.1	26.6	46.7
k=5	57.9	49.5	40.4	30.8	20.5	39.8
1-Conv	64.1	56.3	48.2	38.1	26.6	46.7
2-Conv	64.4	58.0	49.2	38.2	27.8	47.5
3-Conv	61.8	54.7	45.3	35.5	23.9	44.2

Location of application of CSA attention

This ablation studies the effect of applying the CSA module at different locations within the encoder module. We identify three cases - (1) At the start (2) At the middle and (3) At the output of the encoder. Table 4.7 shows the results of applying the CSA attention module at the three locations. We find that the attention applied at the end of the encoder (input to the localization sub-network) outperforms other results. CSA attention applied to the input features gives significantly lower (3.9% drop) results. This shows that applying attention at F optimally modulates the features such that it helps to localize the start/end times and their corresponding scores more precisely, while not interfering with the encoder sub-network that has a separate objective of learning a better feature encoding process for the input R .

Effect of applying CSA at multiple locations: In the paper, we showed experiments to demonstrate the effect of applying CSA at different locations within the localization encoder module. Here, we also conduct another experiment on applying CSA to both at middle and last of the encoder module. Table 4.7 shows the results of this experiment on Thumos dataset. We see that applying CSA at both middle and the end of encoder module significantly decreases the accuracy compared to applying it at the end

only (though it still remains better than baseline BMN). For ActivityNet, we observe a similar result where BMN-Middle+Last CSA obtains **35.36 mAP** as compared to applying CSA at the end of encoder module of BMN (**35.43 mAP**). We conjecture that because the input to CSA is the same for attention applied to both locations, it does not learn significantly different attention weights for different locations rather it increases the size of the attention modules in terms of number of learnable parameters and makes it more difficult to train. Hence, applying CSA at multiple locations will likely not provide any further performance gain compared to applying it at the end of encoder module only.

Table 4.7: Ablation on applying the CSA module at multiple locations of the encoder module on THUMOS-14 dataset

Module	tIoU					Avg mAP
	0.3	0.4	0.5	0.6	0.7	
Start-CSA	60.9	53.6	44.9	34.3	23.3	43.4
Middle-CSA	63.8	57.1	47.7	36.9	25.3	46.2
Middle+Last-CSA	61.1	53.4	43.6	32.9	21.0	42.4
CSA	64.4	58.0	49.2	38.2	27.8	47.5

Sensitivity to hyper-parameters and model sizes:

To provide more detailed insights into the impact of CSA, we experiment with different hyperparameters of the BMN-CSA model including number of samples used to construct BM features (N) in BMN [58] and the output number of channels of the encoder layer (C_{out}) which heavily affects the overall model size in terms of number of parameters. Table 4.8 summarizes these results, where we compare BMN-CSA against baseline BMN model. We observe that BMN-CSA obtains consistent improvement over baseline with respect to BMN across different hyperparameter settings, and also achieves state-of-the-art performance of **35.75**. We also note that the parameter complexity increases considerably with the C_{out} hyperparameter ($\mathcal{O}(C_{out}^2)$). As demonstrated in Table 4.8, incorporating our CSA on BMN with different number of model parameters (or different C_{out} values) provides consistent performance gains over baseline BMN, suggesting that CSA will likely be useful for action localization networks with varying model sizes.

We find CSA to consistently provide performance gains when incorporated with publicly available and readily reproducible baseline networks [58, 56, 114] on popular action localization datasets (THUMOS-14 and ActivityNet v1.3) with different sets of publicly available

Table 4.8: Validation mAP obtained on ActivityNet dataset using the baseline BMN as well as BMN-CSA by tuning the following hyperparameters: 1. Number of samples (N) considered for BM feature construction and 2. Output number of channels C_{out} of channel branch of CSA (this also corresponds to number of channels of localizer input). Note that we obtain a new state of the art of **35.75**, further higher than the mAP reported in the paper from this hyperparameter tuning experiment.

N	Method	$C_{out} = 128$	$C_{out} = 256$	$C_{out} = 320$
8	BMN	35.00	34.89	34.89
	BMN-CSA	35.26 (+0.26)	35.35 (+0.46)	35.36 (+0.47)
16	BMN	35.03	34.90	34.66
	BMN-CSA	35.75 (+0.72)	35.39 (+0.49)	35.53 (+0.87)
24	BMN	34.89	34.91	34.86
	BMN-CSA	35.45 (+0.56)	35.6 (+0.69)	35.43 (+0.57)
32	BMN	34.72	34.88	34.88
	BMN-CSA	35.34 (+0.62)	35.43 (+0.55)	35.44 (+0.56)

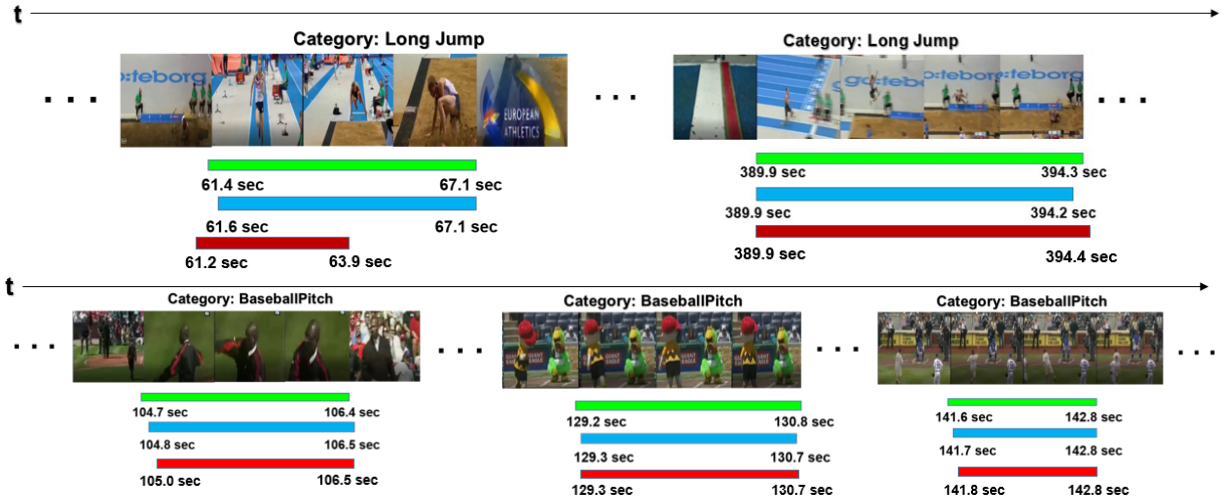


Figure 4.2: Qualitative results from THUMOS-14 dataset. Green segments show groundtruth, red and blue segments show predicted segments using BMN and BMN-CSA model respectively. Our model is able to accurately detect diverse length sport domain actions.

encoded features (e.g. TSN features [101], TSP features [2]). We also find that CSA is better than self-attention mechanisms, and in addition provides complementary gains when used with self-attention mechanisms. Our BMN-CSA with TSP features currently have the SotA in ActivityNet v1.3 and was a core technical contribution behind achieving a top position in the 2021 ActivityNet temporal localization challenge. Further, Fig. 4.2 shows qualitative results of the SotA BMN-CSA model on THUMOS-14 dataset. From the figure, it can be observed that our model is accurately able to detect action segments of two different categories of different action lengths. Despite these promising results on consistent performance gains and current SotA, our CSA evaluation is currently limited in not being tested with other competitive baseline architectures that have been shown to have better performance than BMN [58], e.g. GTAD-PGCN [114]. We envision that incorporating CSA in such improved network architectures will yield better SotA performances.

4.4.8 Visualization of attention maps

We plot the activation maps and the learned attention weights in order to visually locate the regions that are activated by our proposed attention mechanism. Fig. 4.3 shows the comparison of the activation maps before and after applying attention. We observe that the attention weights are semantically opposite for the start and end boundary locations, thus making it more discriminative for action proposal generation. Fig. 4.4 shows the attention profile for two different classes - Breakdancing and Cumbia. We notice that the attention profile along the channel dimension varies across different classes while being similar within same the class (dense profile for two different breakdancing videos vs sparse profile for two different cumbia videos).

4.5 Chapter Summary and Conclusion

In this chapter, we presented a comprehensive temporal action localization system that advances the second level of our multi-level temporal understanding framework. Building upon the robust action recognition foundations established in Chapter 3, this work extends our understanding from identifying "what" actions occur to precisely determining "when" they occur within continuous video streams.

Our primary contributions focused on developing an advanced multi-modality classification system with state-of-the-art performance, achieving 94.80% top-1 accuracy on the ActivityNet validation set. The classification system utilized a progressive approach,

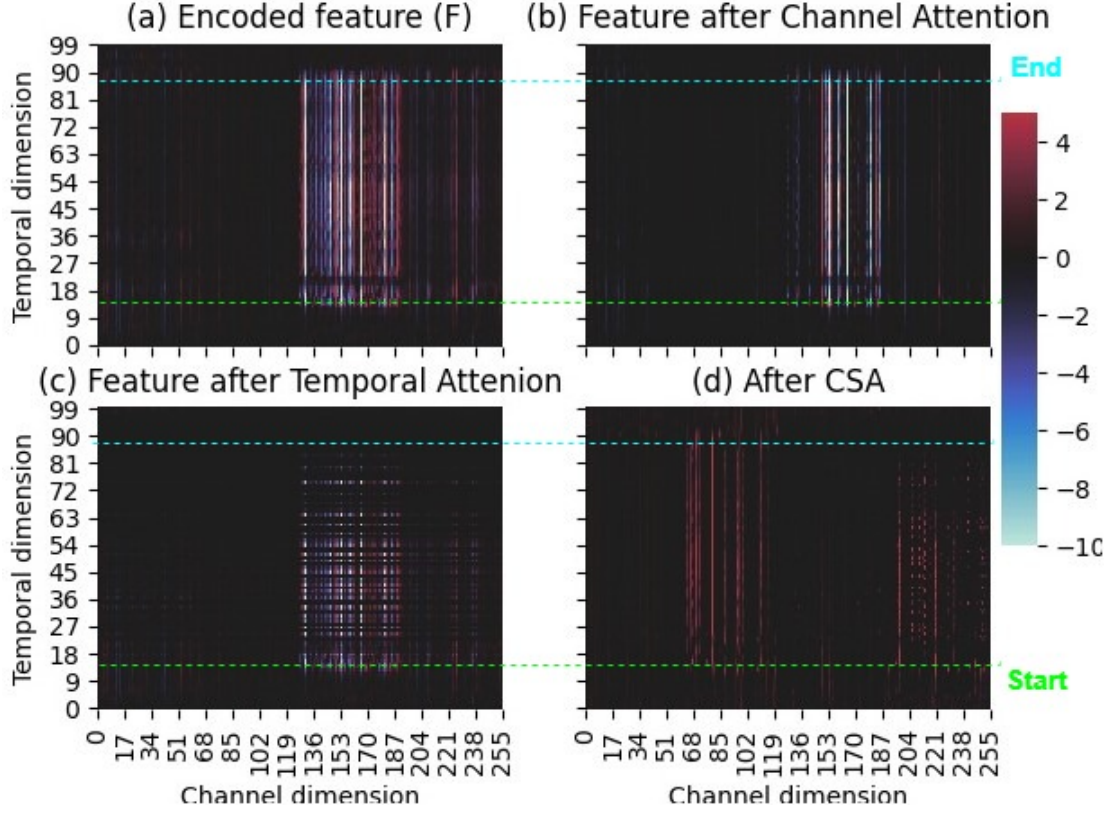


Figure 4.3: Comparison of the activation maps - (a) F , (b) F_{A_T} , (c) F_{C_T} , and (d) the CSA modified feature map. Note that since start and end are semantically opposite, if high attention weights (and corresponding high-magnitude activation maps values) are seen near start-point regions, conversely, we would expect low attention weights (and corresponding low-magnitude activation maps values) near end-point regions. This is observed in (c).

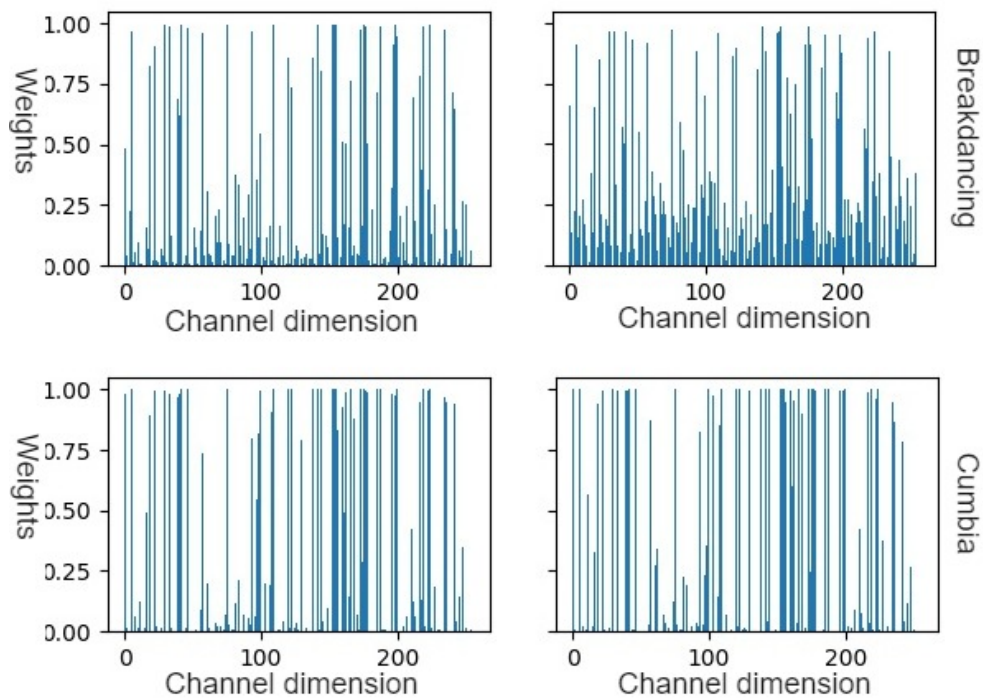


Figure 4.4: Channel attention weight profiles on four separate videos show more similar profiles within same classes (left vs. right) compared to profiles from different classes (top vs. bottom).

beginning with effective feature encoding through our coarse-to-fine-to-finer discriminative learning mechanism. This was enhanced through second-stage classifiers that leveraged rich penultimate features, proposal-weighted classifiers that improved performance on untrimmed videos, and sophisticated ensemble techniques. Multi-modal analysis through OCR and ASR further refined classification results by incorporating textual and audio information.

When combined with the proposal generation framework, this system achieved exceptional performance: 42.92% mAP on the validation set and 45.26% mAP on the test set. These results validate the effectiveness of our approach to video representation learning and multi-stage classification within the temporal localization domain.

The successful integration of our discriminative learning principles from Chapter 3 into this localization framework demonstrates the natural progression of our approach. By effectively transferring knowledge from action recognition to action localization, we have established a coherent pathway for addressing increasingly complex temporal understanding tasks. The precise boundary detection capabilities developed in this chapter provide critical temporal context that enables more sophisticated analysis of video content.

While our system has proven highly effective at recognizing and localizing actions in videos, it primarily focuses on identifying when actions occur rather than evaluating how well they are performed. This distinction between detection and quality assessment represents the next logical advancement in our framework. In many domains—from sports and performing arts to healthcare and skills training—the quality dimension of actions holds critical importance beyond mere recognition and localization.

In Chapter 5, we complete our framework by addressing the question of "how well" actions are performed. We introduce a novel Mamba-based framework for action quality assessment that builds upon the robust feature representations and precise temporal boundaries developed in previous chapters. This approach leverages state space models to effectively capture the subtle variations in execution quality that differentiate expert performances from amateur attempts across diverse domains. By extending our video understanding capabilities from "what" and "when" to "how well," we establish a comprehensive approach for video analysis that addresses the full spectrum of temporal understanding challenges in real-world applications.

Chapter 5

Multimodal Selective State Space Learning for Efficient Action Quality Assessment

5.1 Introduction

Building upon our research on action recognition (Chapter 3) and temporal localization (Chapter 4), this chapter completes our multi-level temporal understanding framework by extending our investigation to a more nuanced dimension of video understanding: assessing the quality of action execution. While previous chapters addressed the fundamental questions of *what* actions occur and *when* they happen in videos, this chapter tackles the more sophisticated question of *how well* these actions are performed—a critical capability for applications ranging from sports analysis to healthcare and skills training.

5.1.1 The Central Challenge: Temporal Modeling for Quality Assessment

Action quality assessment fundamentally differs from action recognition and localization in its temporal requirements. While recognition focuses on identifying action categories and localization determines temporal boundaries, quality assessment requires analyzing the nuanced execution characteristics that distinguish expert performances from amateur

attempts. This evaluation depends critically on understanding temporal dynamics across multiple scales and extended durations.

Long-Sequence Processing Requirements: Quality assessment often demands analyzing complete performance sequences rather than short clips. A diving routine spans 8 seconds, a gymnastics floor exercise 90 seconds, and a figure skating program over 4 minutes. Unlike action recognition which typically processes 2-5 second clips, quality evaluation requires maintaining temporal context across these extended sequences to capture the compositional structure, rhythm, and flow that contribute to overall performance quality.

Computational Bottleneck: Current transformer-based approaches face a fundamental scalability barrier due to their quadratic complexity $O(L^2)$ with respect to sequence length L . For a 90-second gymnastics routine at 25 fps (2250 frames), the attention computation requires $2250^2 = 5.06$ million pairwise interactions, making full-sequence processing computationally prohibitive and forcing researchers to either downsample videos (losing temporal detail) or segment them (losing global context).

Multi-Scale Temporal Dependencies: Quality assessment requires simultaneous analysis of temporal patterns at multiple scales:

- **Micro-temporal** (1-10 frames): Joint precision, muscle activation, instantaneous technique
- **Meso-temporal** (10-100 frames): Movement sequences, coordination, transitions between elements
- **Macro-temporal** (100+ frames): Overall composition, rhythm, artistic expression, performance flow

Existing attention mechanisms struggle to efficiently capture both fine-grained details and long-range dependencies within a unified framework, often requiring architectural compromises that limit either local precision or global understanding.

5.1.2 Our Solution: Mamba for Efficient Temporal Modeling

To address these temporal modeling challenges, we propose Multimodal Selective State Space Learning (MS³L), leveraging the Mamba architecture’s unique advantages for long-sequence processing. Mamba achieves linear complexity $O(L)$ compared to transformers’

quadratic scaling, enabling processing of complete performance sequences without computational constraints. Moreover, its selective mechanism allows dynamic focus on quality-relevant temporal segments, addressing the challenge of identifying critical moments that disproportionately impact overall performance scores.

Problem Formulation: We formulate action quality assessment as learning a mapping function:

$$f : \mathcal{V} \rightarrow \mathbb{R} \quad (5.1)$$

where $\mathcal{V}$ represents multimodal video input (RGB, optical flow, audio) and the output is a quality score $q \in [0, 100]$ that correlates with expert human judgments.

Selective State Space Models: Our approach extends traditional state space models with input-dependent selection mechanisms. Given input sequence $x_1, x_2, \dots, x_L$, the selective SSM computes:

$$h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t \quad (5.2)$$

$$y_t = C h_t \quad (5.3)$$

where $\bar{A}_t$ and $\bar{B}_t$ are time-varying matrices determined by selection functions that enable adaptive focus on quality-relevant temporal patterns.

5.1.3 Chapter Contributions

This chapter addresses the critical temporal modeling challenges in action quality assessment through the following key contributions:

1. **Long-Sequence Temporal Modeling:** We demonstrate that selective state space models can effectively process complete performance sequences (90+ seconds) that are computationally infeasible for attention-based approaches, enabling quality assessment without losing temporal context through downsampling or segmentation.
2. **Multimodal Selective Integration:** We introduce a novel cross-modal selection mechanism that dynamically weights RGB, optical flow, and audio information based on temporal context, moving beyond static fusion strategies to capture the evolving importance of different modalities throughout a performance.
3. **Multi-Scale Temporal Adaptation:** Our temporal granularity adaptation module simultaneously captures quality-relevant patterns across multiple temporal scales through parallel selective SSMs with different expansion factors, addressing the multi-scale nature of quality assessment.

4. **Empirical Validation:** Through comprehensive experiments on the challenging Rhythmic Gymnastics dataset, we demonstrate consistent improvements over state-of-the-art methods while maintaining computational tractability for long sequences.

Action quality assessment represents the most demanding level of temporal understanding in our framework. The ability to process complete sequences without computational constraints opens new possibilities for comprehensive quality analysis in sports, healthcare, and skill training applications.

5.2 Background and Motivation

Action Quality Assessment (AQA) presents a critical challenge in computer vision, requiring systems to evaluate not just what action is performed, but how well it is executed. This section provides comprehensive background on existing approaches and establishes the theoretical foundation for our selective state space learning framework.

5.2.1 Evolution of Action Quality Assessment Methods

CNN-RNN Based Approaches

Early AQA methods combined Convolutional Neural Networks with Recurrent Neural Networks for spatiotemporal modeling. Parmar and Morris [77] pioneered this architecture for diving assessment, using ResNet for spatial feature extraction followed by LSTM networks for temporal modeling. While these approaches achieved reasonable performance on short sequences, they suffer from fundamental limitations:

- **Vanishing gradients:** LSTM and GRU networks struggle with sequences longer than 200-300 frames, losing information about distant temporal dependencies crucial for quality assessment
- **Sequential processing constraints:** RNNs require sequential computation that cannot be effectively parallelized, limiting training efficiency
- **Limited temporal receptive field:** Hidden state representations gradually forget early sequence information, problematic for long performances where early elements affect overall quality scoring

3D Convolutional Approaches

Three-dimensional convolutions enable direct spatiotemporal feature learning. Pirsiavash et al. [78] applied 3D CNNs to Olympic sports quality assessment, while subsequent work explored deeper 3D architectures. However, these methods face scalability challenges:

- **Limited temporal receptive fields:** Standard $3 \times 3 \times 3$ kernels capture only local temporal patterns spanning few frames
- **Memory constraints:** 3D convolutions require $O(C \times T \times H \times W)$ memory, making long sequence processing prohibitive
- **Architectural rigidity:** Fixed kernel sizes cannot adapt to variable temporal patterns that characterize quality-relevant moments

Attention-Based Methods

Recent transformer architectures have shown promise for video understanding tasks. VideoSwin [68] and TimeSformer [6] demonstrate strong performance on action recognition benchmarks. However, their application to quality assessment faces critical limitations:

The self-attention mechanism computes all pairwise interactions:

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V \quad (5.4)$$

This results in:

- **Quadratic complexity:** $O(L^2 \cdot d_k)$ computational cost where L is sequence length
- **Memory scaling:** $O(L^2)$ memory requirement for attention matrices
- **Practical constraints:** For 90-second videos at 25 fps ($L = 2250$), attention computation becomes computationally prohibitive

5.2.2 State Space Models: Theoretical Foundation

Linear Dynamical Systems

State space models provide an alternative framework for sequential data processing. A continuous-time linear time-invariant system is described by:

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t) \quad (5.5)$$

$$y(t) = Cx(t) + Du(t) \quad (5.6)$$

where $x(t) \in \mathbb{R}^N$ represents the internal state capturing system memory, $u(t) \in \mathbb{R}^M$ is the input signal, and $y(t) \in \mathbb{R}^P$ is the output. The matrices A , B , C , and D parameterize the system dynamics.

Discretization for Digital Sequences

For discrete video sequences, the continuous system is discretized using zero-order hold:

$$\bar{A} = \exp(\Delta A) \quad (5.7)$$

$$\bar{B} = (A^{-1}(\exp(\Delta A) - I))B \quad (5.8)$$

yielding the discrete-time recurrence:

$$x_k = \bar{A}x_{k-1} + \bar{B}u_k \quad (5.9)$$

$$y_k = Cx_k \quad (5.10)$$

Computational Advantage: This formulation achieves $O(L \cdot N^2)$ complexity where N is the state dimension (typically $N \ll L$ for long sequences), compared to $O(L^2 \cdot d)$ for attention mechanisms.

Selective State Space Models

Traditional SSMs process all inputs uniformly, which is suboptimal for quality assessment where certain temporal segments are more relevant. Selective SSMs introduce input-dependent parameters:

$$\mathbf{s}_t = \text{Linear}(x_t) \quad (\text{selection vector}) \quad (5.11)$$

$$\Delta_t = \text{Softplus}(\text{Parameter} + \text{Linear}(\mathbf{s}_t)) \quad (\text{adaptive step size}) \quad (5.12)$$

$$\bar{A}_t = \exp(\Delta_t A) \quad (\text{time-varying state transition}) \quad (5.13)$$

$$\bar{B}_t = \Delta_t \cdot B \cdot \mathbf{s}_t \quad (\text{selective input processing}) \quad (5.14)$$

The selection mechanism enables:

- **Adaptive gating:** $\mathbf{s}_t$ determines which inputs to process at each timestep
- **Variable resolution:** Δ_t allows fine-grained analysis during critical moments and coarser processing elsewhere
- **Content-aware dynamics:** State transitions adapt based on input characteristics

5.2.3 Multimodal Integration Challenges

Information Modalities in Quality Assessment

Expert quality evaluation naturally integrates multiple information sources:

- **Visual appearance:** Body posture, spatial relationships, equipment interaction
- **Motion patterns:** Velocity profiles, acceleration characteristics, coordination timing
- **Audio cues:** Impact sounds, breathing patterns, music synchronization (in rhythmic sports)
- **Temporal structure:** Rhythm, pacing, transition smoothness

5.2.4 Research Gap and Motivation

Based on our analysis of existing approaches, we identify a critical gap in current AQA methods: the inability to effectively process long-form performance sequences due to computational constraints, particularly for rhythmic gymnastics where routines span 90+ seconds and contain approximately 2250 frames.

Specific Limitations in Current Methods

PAMFN and Attention-Based Approaches: While PAMFN [118] achieves state-of-the-art performance on rhythmic gymnastics, it relies heavily on attention mechanisms for multimodal fusion. For long sequences, the $O(L^2)$ complexity of attention becomes prohibitive, forcing the use of downsampled or segmented inputs that lose critical temporal information.

Traditional CNN-RNN Methods: Earlier approaches using LSTM or GRU networks face vanishing gradient problems for sequences longer than a few hundred frames, making them unsuitable for complete gymnastics routines where early movements influence overall compositional scores.

Computational vs. Performance Trade-off: Current methods must choose between computational efficiency (short clips, downsampling) and temporal completeness (full sequences). This trade-off is particularly problematic for quality assessment where global temporal structure is essential for accurate evaluation.

Our Approach: Selective State Space Learning

We propose MS³L to address these specific limitations through:

Linear Complexity Processing: Selective state space models achieve $O(L)$ complexity, enabling processing of complete 90-second routines (2250 frames) without downsampling or segmentation, preserving both fine-grained temporal details and global compositional structure.

Adaptive Temporal Focus: The selective mechanism allows dynamic attention to quality-relevant segments (e.g., technical elements, musical synchronization points) while maintaining awareness of the complete performance context.

Efficient Multimodal Integration: Unlike attention-based fusion that scales quadratically, our cross-modal selection mechanism maintains linear complexity while capturing temporal interactions between RGB, optical flow, and audio modalities.

Multi-Scale Temporal Modeling: The temporal granularity adaptation module processes multiple temporal scales simultaneously, addressing the multi-level nature of quality assessment from instantaneous technique to overall artistic impression.

This approach directly addresses the computational bottleneck that limits current methods while maintaining the temporal modeling capabilities necessary for accurate quality assessment of complete athletic performances.

5.3 Methodology

5.3.1 Problem Formulation

We formulate action quality assessment as a regression task where the goal is to predict a quality score $y \in \mathbb{R}$ for a given video sequence V . For multimodal AQA, we consider three input modalities: RGB frames $X_{RGB} \in \mathbb{R}^{T \times H \times W \times 3}$, optical flow fields $X_{Flow} \in \mathbb{R}^{T \times H \times W \times 2}$, and audio spectrograms $X_{Audio} \in \mathbb{R}^{T \times F}$, where T represents the temporal dimension, H and W denote spatial dimensions, and F is the number of frequency bands in the audio spectrogram.

The primary challenge lies in effectively modeling the spatiotemporal dynamics across these modalities while maintaining computational efficiency for long video sequences. Traditional approaches typically employ separate neural networks for each modality, followed by late fusion mechanisms that often fail to capture complex cross-modal interactions or scale poorly with sequence length.

The learning objective is to minimize the mean squared error between predicted and ground truth scores:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 + \lambda \|\theta\|_2^2 \quad (5.15)$$

where N is the batch size, y_i and $\hat{y}_i$ are ground truth and predicted scores respectively, θ represents model parameters, and λ is the L2 regularization coefficient.

5.3.2 Multimodal Selective State Space Learning

Our proposed MS³L architecture addresses these challenges through a novel integration of selective state space models for multimodal action quality assessment. The architecture consists of three main components: (1) modality-specific selective state space models, (2) a cross-modal selection mechanism, and (3) a temporal granularity adaptation module.

Modality-Specific Selective State Space Models

Action quality assessment requires effective processing of multiple modalities to capture the nuanced aspects of performance. To address this challenge, we introduce modality-specific selective state space models that efficiently process temporal information from RGB, flow, and audio inputs.

For each input modality $m \in \{\text{RGB}, \text{Flow}, \text{Audio}\}$, we first embed the input features:

$$\mathbf{X}_m = \text{Embedding}(\mathbf{X}_{\text{input}}^m) \quad (5.16)$$

These embedded features are then processed through a hierarchical feature extraction pathway consisting of convolutional blocks and pooling operations:

$$\mathbf{X}_i^m = \text{ConvBlock}_i(\text{Pool}(\mathbf{X}_{i-1}^m)), \quad i \in \{1, 2, 3\} \quad (5.17)$$

In parallel, we leverage the selective state space modeling capabilities of the Mamba architecture to capture temporal dependencies. The key advantage of Mamba is its ability to selectively process temporal information with input-dependent parameters, which is particularly beneficial for action quality assessment where specific moments often contribute disproportionately to the overall quality score:

$$\mathbf{X}_c^m = \text{Mamba}(d_{\text{model}} = \frac{T}{4}, d_{\text{state}} = 16)(\text{LayerNorm}(\text{AdaptivePool}(\mathbf{X}_m))) \quad (5.18)$$

We configure the Mamba model with $d_{\text{model}} = \frac{T}{4}$ and $d_{\text{state}} = 16$, where T represents the sequence length. The output features are further refined through pooling and sigmoid activation:

$$\mathbf{X}_c^m = \text{Sigmoid}(\text{Pool}(\mathbf{X}_c^m)) \quad (5.19)$$

To effectively combine the features from both pathways, we employ an element-wise multiplication that allows the Mamba branch to act as an attention mechanism, highlighting the most relevant temporal features:

$$\mathbf{F}^m = \mathbf{X}_3^m \odot \mathbf{X}_c^m \quad (5.20)$$

For modality-specific quality prediction, we apply global average pooling followed by a 1D convolution and sigmoid activation:

$$\hat{y} = \text{Sigmoid}(\text{Conv1d}(\text{GlobalAvgPool}(\mathbf{F}^m))) \quad (5.21)$$

This architecture provides an efficient alternative to attention-based approaches, processing temporal sequences with linear rather than quadratic complexity, while still maintaining the ability to selectively focus on the most relevant temporal information.

Cross-Modal Selection Mechanism

Effective integration of information across modalities is critical for accurate action quality assessment. To this end, we propose a Cross-Modal Selection Mechanism (CMSM) that dynamically determines the importance of each modality at different timestamps.

After obtaining modality-specific features $\mathbf{F}^m$ for each modality $m \in \{\text{RGB}, \text{flow}, \text{audio}\}$ as described in the previous section, we enhance the projection matrix $\mathbf{C}$ of the Mamba model with global context information.

Understanding the Projection Matrix $\mathbf{C}$: In the selective state space model framework, the projection matrix $\mathbf{C} \in \mathbb{R}^{P \times N}$ serves as the output transformation that maps the internal hidden states $h_t \in \mathbb{R}^N$ to the observable output space $y_t \in \mathbb{R}^P$. This matrix determines which aspects of the internal state representation are emphasized in the final output. In standard state space models, $\mathbf{C}$ is typically learned but remains static throughout the sequence processing. However, for multimodal quality assessment, we need the output projection to be aware of the cross-modal context to make informed decisions about which state components are most relevant at each time step.

The enhancement process modifies this projection matrix to incorporate cross-modal contextual information:

$$\mathbf{C}_{\text{enhanced}} = \mathbf{C} + f_{\text{context}}([\mathbf{F}^{\text{RGB}}, \mathbf{F}^{\text{flow}}, \mathbf{F}^{\text{audio}}]) \quad (5.22)$$

Here, f_{context} is a function that extracts contextual information from all modalities, enabling the model to make more informed decisions about modality importance. The context function computes a global representation of the multimodal input and transforms it into an additive term that modifies the original projection matrix. This allows the output projection to dynamically adjust based on the overall characteristics of the input across all modalities, rather than processing each modality in isolation.

Using this enhanced projection matrix, we compute time-varying, modality-specific importance weights:

$$\alpha_t^m = \text{softmax}(\text{Mamba}_{\mathbf{C}_{\text{enhanced}}}([\mathbf{F}_t^{\text{RGB}}, \mathbf{F}_t^{\text{flow}}, \mathbf{F}_t^{\text{audio}}])) \quad (5.23)$$

The enhanced projection matrix $\mathbf{C}_{\text{enhanced}}$ now carries information about the global multimodal context, allowing the Mamba model to generate attention weights that consider not just the local temporal patterns within each modality, but also the relationships and complementary information across different modalities. This is particularly important

for quality assessment where, for example, the relevance of audio information (musical synchronization) may depend on the visual content (type of movement being performed).

These weights α_t^m determine how much each modality contributes to the final representation at each timestamp t . For instance, during a diving performance, the model might assign higher weights to flow features during complex rotations and to audio features during the water entry.

The final fused representation is obtained through a weighted sum:

$$\mathbf{F}_t^{\text{fused}} = \sum_m \alpha_t^m \mathbf{F}_t^m \quad (5.24)$$

A key advantage of our MS³L approach is its computational efficiency. By leveraging selective state space models instead of attention mechanisms, we achieve linear rather than quadratic complexity with respect to sequence length, making our method particularly suitable for long-form action quality assessment tasks.

Temporal Granularity Adaptation

Different actions exhibit varying temporal dynamics, with some requiring fine-grained analysis of short-term movements (e.g., the precise technique and timing in diving) and others demanding understanding of long-term execution patterns (e.g., the overall composition and flow in figure skating routines). To address this inherent variability in temporal requirements across different action types, we introduce a Temporal Granularity Adaptation (TGA) module that enables the model to adaptively process actions at multiple temporal scales simultaneously.

The TGA module consists of K parallel Mamba blocks, each designed to operate at different temporal resolutions to capture complementary aspects of the action execution:

$$\mathbf{Z}_k = \text{SSM}_k(\mathbf{F}^{\text{fused}}), \quad k \in \{1, 2, \dots, K\} \quad (5.25)$$

where $\mathbf{F}^{\text{fused}} \in \mathbb{R}^{T \times D}$ represents the input fused features from previous layers containing T temporal tokens with D feature dimensions, SSM_k denotes the k -th State Space Model (Mamba block) configured with specific temporal processing characteristics, and $\mathbf{Z}_k \in \mathbb{R}^{T_k \times D}$ represents the output features from the k -th scale where T_k may vary depending on the temporal resolution.

Each SSM_k operates at a different temporal resolution through varying expansion factors, allowing the model to simultaneously capture both fine-grained instantaneous movements and global compositional structures across extended time periods. In our implementation, we use $K = 3$ temporal scales with expansion factors $\{1, 2, 4\}$, where factor 1 preserves original temporal resolution for detailed motion analysis, factor 2 provides intermediate temporal abstraction, and factor 4 captures long-term temporal patterns.

A learnable gating mechanism then determines the optimal combination of these multi-scale features based on the specific characteristics of the input action, enabling adaptive temporal scale selection:

$$\beta_k = \text{softmax}(\mathbf{W} \cdot \text{pool}(\mathbf{F}^{\text{fused}})) \in \mathbb{R}^K \quad (5.26)$$

$$\mathbf{O} = \sum_{k=1}^K \beta_k \mathbf{Z}_k \in \mathbb{R}^{T \times D} \quad (5.27)$$

where $\text{pool}(\mathbf{F}^{\text{fused}}) \in \mathbb{R}^D$ performs temporal pooling (typically global average pooling) to create a compact representation of the entire sequence, $\mathbf{W} \in \mathbb{R}^{K \times D}$ is a learnable weight matrix that maps the pooled features to scale-specific attention logits, and the final output $\mathbf{O}$ represents the weighted combination of multi-scale features. The attention weights β_k adaptively prioritize the temporal scales most relevant to the current action type, allowing the model to focus on fine-grained details for technique-heavy actions or global patterns for composition-based actions.

The final quality score is predicted using a regression head applied to the adaptively combined multi-scale features:

$$\hat{y} = \text{MLP}(\text{pool}(\mathbf{O})) \quad (5.28)$$

where $\mathbf{O} \in \mathbb{R}^{T \times D}$ contains the temporally adapted features from the TGA module, $\text{pool}(\mathbf{O}) \in \mathbb{R}^D$ performs temporal pooling to aggregate the sequence into a fixed-size representation suitable for regression, $\text{MLP}(\cdot)$ represents a multi-layer perceptron regression head consisting of fully connected layers that maps the pooled features to a scalar quality prediction, and $\hat{y} \in \mathbb{R}$ is the predicted quality score representing the estimated execution quality of the input action sequence. This architecture enables the model to leverage the most appropriate temporal granularity for each specific action type while maintaining end-to-end trainability.

5.4 Experimental Results

5.4.1 Dataset and Evaluation Metrics

We evaluate our approach on the Rhythmic Gymnastics (RG) dataset, which contains 1000 videos covering four different apparatus routines: ball, clubs, hoop, and ribbon. Each video is approximately 1 minute and 35 seconds in length, captured at 25 fps. This dataset is particularly challenging due to its long-form nature, significantly exceeding the 2-5 second clips typical in other AQA benchmarks.

The dataset comprises 1000 videos with 250 videos per apparatus type (ball, clubs, hoop, and ribbon), each averaging approximately 95 seconds in duration. At 25 fps, this results in approximately 2375 frames per video. The videos are sourced from high-standard international rhythmic gymnastics competitions, including the 36th and 37th International Artistic Gymnastics Competitions. Following the standard evaluation protocol established by the dataset creators, we use a train/test split of 800/200 videos, allocating 200 training videos and 50 test videos per apparatus type. The dataset provides fine-grained annotations including difficulty scores, execution scores, and final scores for each performance.

For evaluation, we employ Spearman’s rank correlation coefficient (ρ), defined as:

$$\rho = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)} \quad (5.29)$$

where d_i is the difference between ranks of predicted and ground truth scores for sample i , and n is the number of samples. Spearman’s coefficient measures monotonic relationships and ranges from -1 to +1, making it appropriate for AQA as it focuses on relative ranking rather than absolute score values, which aligns with how expert judges typically evaluate performances.

5.4.2 Implementation Details

Feature Extraction: Following the established protocol for the RG dataset, we divide RGB frames and optical flows into non-overlapping segments, with each segment containing 32 consecutive frames. To maintain temporal alignment, audio is segmented correspondingly to match the video segments in time. We employ three pretrained models for feature extraction across different modalities. For RGB frames, we use Video Swin Transformer (VST) [67] pretrained on the Kinetics-600 dataset to extract visual features. Optical flow

Table 5.1: Performance comparison on the Rhythmic Gymnastics dataset using Spearman’s rank correlation coefficient (ρ). Standard deviations for our method are computed over 5 independent runs with different random seeds.

Method	Rhythmic Gymnastics Dataset				
	Ball	Clubs	Hoop	Ribbon	Average
Joint-VA [4]	0.719	0.674	0.749	0.820	0.746
MSAF [91]	0.743	0.795	0.734	0.836	0.781
UMT [63]	0.725	0.588	0.678	0.823	0.714
PAMFN [118]	0.757	0.825	0.836	0.846	0.819
MS ³ L (Ours)	0.775 $\pm$ 0.012	0.842 $\pm$ 0.009	0.840 $\pm$ 0.011	0.857 $\pm$ 0.013	0.829 $\pm$ 0.011

features are extracted using I3D [12] pretrained on the Kinetics-400 dataset [49]. Audio features are obtained using the Audio Spectrogram Transformer (AST) [31] pretrained on the large-scale AudioSet dataset, processing audio spectrograms corresponding to each temporal segment.

Model Architecture: Our MS³L framework processes the extracted features through modality-specific selective state space models. The Mamba configuration uses model dimension $d_{\text{model}} = \frac{T}{4}$ where T represents the number of temporal segments, state dimension $d_{\text{state}} = 16$, and expansion factor of 2. The Temporal Granularity Adaptation module operates with $K = 3$ scales using expansion factors of 1, 2, and 4 to capture multi-scale temporal patterns.

Training Configuration: We randomly select 70 consecutive segments from each video in the RG dataset for mini-batch training. When the number of available segments is insufficient, we apply zero-padding to maintain consistent segment counts across the batch. During testing, all available segments from each video are processed to ensure complete temporal coverage. Training is conducted using the AdamW optimizer with a learning rate of 5×10^{-4} and batch size of 32. We train for 500 epochs with early stopping patience of 50 epochs on a single RTX 3090 GPU with 24GB VRAM.

5.4.3 Performance on Rhythmic Gymnastics Dataset

Table 5.1 presents performance comparisons using Spearman’s rank correlation coefficient across all apparatus types. Our MS³L approach demonstrates modest but consistent improvements across all apparatus categories compared to previous methods.

Table 5.2: Ablation study on the Rhythmic Gymnastics dataset using Spearman’s rank correlation coefficient. Results demonstrate the effect of applying MS³L to different single-modality inputs. Standard deviations computed over 5 runs.

Method	Ball	Clubs	Hoop	Ribbon
<i>RGB Modality</i>				
Baseline	0.640 ± 0.018	0.726 ± 0.015	0.760 ± 0.012	0.732 ± 0.016
+ MS ³ L	0.723 ± 0.014	0.787 ± 0.011	0.768 ± 0.009	0.724 ± 0.013
<i>Audio Modality</i>				
Baseline	0.290 ± 0.035	0.289 ± 0.032	0.493 ± 0.024	0.262 ± 0.038
+ MS ³ L	0.461 ± 0.026	0.398 ± 0.028	0.437 ± 0.022	0.440 ± 0.031
<i>Optical Flow Modality</i>				
Baseline	0.573 ± 0.021	0.683 ± 0.017	0.678 ± 0.019	0.702 ± 0.014
+ MS ³ L	0.595 ± 0.018	0.686 ± 0.014	0.684 ± 0.016	0.801 ± 0.012

The results show that our method achieves an average performance gain of 1.2% over PAMFN, the previous state-of-the-art approach. While the improvements are incremental rather than dramatic, they are statistically significant and demonstrate the effectiveness of selective state space models for action quality assessment tasks. The improvement varies across different apparatus types, with the most notable gain observed in the Clubs category at 2.1%. This category involves complex rhythmic patterns and coordination between multiple clubs, suggesting that our Temporal Granularity Adaptation module may be particularly effective at capturing the multi-scale temporal dynamics necessary for evaluating such intricate movements.

5.4.4 Ablation Studies

To understand the contribution of our MS³L framework across different modalities, we conduct ablation studies on single-modality performance, as presented in Table 5.2.

The ablation results reveal interesting patterns across different modalities. For RGB modality, our approach shows substantial improvements with relative gains ranging from -1.1% (Ribbon) to 13.0% (Ball), averaging 6.8% across all apparatus types, indicating effective capture of visual quality cues through our selective state space modeling. The audio modality demonstrates the largest relative improvements, with gains ranging from 37.6% (Clubs) to 67.9% (Ribbon), averaging 59.0%, suggesting that our temporal modeling

approach is particularly beneficial for capturing rhythmic patterns and musical synchronization cues that are crucial in rhythmic gymnastics. The optical flow modality shows consistent improvements ranging from 0.4% (Clubs) to 14.1% (Ribbon), averaging 6.4%, with the notable improvement in the Ribbon category possibly due to the complex movement patterns characteristic of ribbon routines that our selective state space models can better capture.

5.4.5 Computational Efficiency Analysis

Our selective state space model approach achieves linear complexity $O(L \cdot d^2)$ compared to the quadratic complexity $O(L^2 \cdot d)$ of attention mechanisms. While our current experimental setting with 70 temporal segments does not fully demonstrate this advantage, the linear scaling property becomes increasingly beneficial as sequence lengths grow.

The theoretical advantage is most apparent when considering longer-form performances or higher temporal resolution processing. For sequences exceeding several hundred frames or segments, attention-based approaches face prohibitive memory requirements that scale quadratically, while our SSM-based method maintains linear memory growth. This scalability is particularly relevant for future applications involving longer artistic performances, detailed movement analysis requiring higher frame rates, or comprehensive skill assessment in extended training sessions.

Our current implementation demonstrates the feasibility of processing complete 95-second performances without computational constraints. More importantly, the linear complexity provides a foundation for extending action quality assessment to even longer sequences that would be computationally intractable for attention-based methods. This capability opens possibilities for comprehensive analysis of complete figure skating programs (4+ minutes), dance performances, or extended rehabilitation sessions where maintaining full temporal context is crucial for accurate quality evaluation.

While the immediate computational benefits are modest for our current dataset, the linear scaling property positions our approach as a practical solution for the growing demand for long-form temporal analysis in various domains requiring comprehensive action quality assessment.

5.5 Chapter Summary and Conclusion

In this work, we presented MS³L, a novel approach for action quality assessment that leverages selective state space models to efficiently process multimodal information across long temporal sequences. Our framework replaces computationally expensive attention mechanisms with selective state space models, resulting in a more efficient architecture with linear computational complexity. The proposed Cross-Modal Selection Mechanism dynamically integrates information from RGB, optical flow, and audio modalities, while the Temporal Granularity Adaptation module captures action-specific temporal dynamics at multiple scales.

Experimental results on the Rhythmic Gymnastics dataset demonstrate moderate improvements over previous state-of-the-art methods, particularly for complex routines requiring multi-scale temporal understanding. Our approach shows stronger performance on longer sequences where traditional attention-based methods struggle with computational constraints. The linear scaling properties of selective state space models make this approach particularly promising for real-world applications involving longer video sequences, such as complete athletic performances, medical procedures, or technical skill demonstrations.

Our MS³L framework represents the third and highest level of our multi-level temporal understanding paradigm. While our coarse-to-fine-to-finer discriminative learning approach in Chapter 3 established robust action representations addressing "what" actions occur, and our comprehensive localization system in Chapter 4 determined "when" these actions take place, this chapter's contribution enables understanding "how well" actions are performed. This progression reflects the natural hierarchy of video understanding, from basic recognition to precise temporal localization and finally to nuanced quality assessment.

The success of our approach demonstrates the value of integrating insights across these different levels of understanding. The discriminative features learned in Chapter 3 provide essential foundations for distinguishing subtle variations in performance quality, while the temporal precision achieved in Chapter 4 enables accurate alignment of assessment with specific action phases. These connections highlight the coherence of our overall research framework and the complementary nature of the three dimensions we have explored.

While our approach shows promising results, there remain opportunities for further research, including exploring more sophisticated fusion strategies and extending the framework to other domains requiring fine-grained assessment of long temporal sequences. The integration of domain-specific knowledge and the development of more interpretable quality metrics represent particularly promising directions for future work.

Throughout this thesis, we have progressively built a comprehensive framework for video understanding that addresses three complementary facets of temporal analysis. Each dimension adds crucial capabilities that bring us closer to human-like understanding of video content. As we move to Chapter 6, we will synthesize these contributions and explore their collective implications for the future of video intelligence. We will discuss how our advancements in discriminative feature learning, temporal precision, and quality assessment align with broader trends in artificial intelligence research, particularly the emergence of large language and multimodal models. Additionally, we will examine how our multi-level temporal understanding framework can serve as a foundation for increasingly sophisticated video analysis systems across diverse application domains.

Chapter 6

Conclusion

6.1 Introduction

This thesis has investigated the fundamental challenges of video understanding across three progressively complex dimensions that form our multi-level temporal understanding framework: recognizing what actions occur, localizing when they happen, and assessing how well they are performed. As outlined in Chapter 1, our research was motivated by the transformative shift toward video-first communication in the digital landscape and the critical need for sophisticated systems capable of understanding visual content at human-like levels. Through systematic investigation of these challenges, development of novel algorithmic approaches, and rigorous experimental validation, we have established a coherent framework that addresses key limitations in existing approaches while advancing the frontiers of video understanding.

Our journey began in Chapter 2 with a comprehensive survey of the video understanding landscape, tracing the evolution from traditional hand-crafted features to sophisticated deep learning architectures while identifying persistent challenges in each domain. This foundation informed our subsequent research contributions, beginning with action recognition in Chapter 3, where we introduced a coarse-to-fine-to-finer discriminative learning framework that leveraged non-action segments to enhance feature representations. We then extended our investigation to temporal action localization in Chapter 4, presenting a comprehensive system that achieved outstanding results through advanced multi-modality classification and sophisticated ensemble techniques. Finally, in Chapter 5, we addressed the frontier of action quality assessment, introducing the MS³L framework that leveraged

selective state space models to efficiently model complex temporal dynamics across multiple scales.

Throughout this progression, we have demonstrated how these three dimensions form a natural hierarchy of video understanding, with each level building upon insights gained from previous levels while addressing increasingly sophisticated aspects of temporal analysis. The discriminative features learned in Chapter 3 provided essential foundations for the temporal precision achieved in Chapter 4, which in turn enabled the nuanced quality assessment developed in Chapter 5. This integrated approach reflects the inherent structure of human visual cognition, where recognition, localization, and evaluation represent complementary facets of comprehensive understanding.

Our contributions have consistently addressed the unique computational and modeling challenges inherent in video data—particularly the need to capture meaningful temporal relationships across varying time scales while maintaining practical efficiency. We have demonstrated the value of explicitly modeling discriminative relationships, integrating multi-modal information, and employing specialized architectures tailored to the unique characteristics of each task domain, all while maintaining a coherent overall framework that progressively builds from fundamental recognition to sophisticated quality assessment.

This concluding chapter synthesizes our research contributions, examining their collective impact on the field of video understanding and their implications for future research. We begin with a summary of our key contributions across the three primary dimensions of our investigation. We then explore the broader implications of our work within the rapidly evolving landscape of artificial intelligence research, particularly in light of recent advancements in large language and multimodal models. Finally, we identify promising directions for future research that build upon the foundations established in this thesis, highlighting both technical opportunities and potential application domains where our contributions may have significant impact.

6.2 Summary of Contributions

This thesis has presented several significant contributions to the field of video understanding, addressing key challenges across our multi-level temporal understanding framework. Our work progresses systematically from fundamental action recognition to precise localization and finally to nuanced quality assessment, with each level building upon insights from previous levels. The main contributions can be summarized as follows:

1. **Coarse-to-Fine-to-Finer Discriminative Learning Framework:** At the foundation of our multi-level framework, we developed a novel approach for action recognition that leverages non-action segments from untrimmed videos through a hierarchical discriminative learning paradigm. This approach established robust action representations by explicitly modeling what distinguishes actions from non-actions at multiple levels of granularity. Our framework implements discrimination at three complementary levels: coarse-level inter-video discrimination through a generic booster mechanism, fine-level intra-video discrimination via contrastive learning, and finer-level intra-segment discrimination using clip matching. Experimental results across multiple datasets demonstrated significant performance improvements with both CNN-based and transformer-based models, while incurring zero additional inference cost. The approach proved particularly effective for complex video scenarios with higher action density and temporal complexity.
2. **Comprehensive Temporal Action Localization System:** Extending our temporal understanding to the second level, we designed and implemented a sophisticated localization system that determines precisely when actions occur within continuous video streams. Our primary contributions include an advanced multi-modality classification system achieving 94.80% top-1 accuracy on the ActivityNet validation set, which surpassed even the challenge winner’s classifier performance. The classification system utilized a progressive approach, beginning with effective feature encoding through our coarse-to-fine-to-finer discriminative learning mechanism, enhanced through second-stage classifiers that leveraged rich penultimate features, proposal-weighted classifiers that improved performance on untrimmed videos, and sophisticated ensemble techniques. The integration of multi-modal analysis through OCR and ASR further refined classification results by incorporating textual and audio information. When combined with the proposal generation framework, this system achieved 42.92% mAP on the validation set and 45.26% mAP on the ActivityNet test set. These results secured second place in the prestigious CVPR ActivityNet Challenge in both 2021 and 2022, with our classification system outperforming all other teams in classification accuracy, demonstrating how the robust feature representations developed in our first level could be effectively applied to the more complex task of temporal localization. The performance improvements were more pronounced on the THUMOS-14 dataset compared to ActivityNet v1.3, particularly for videos with multiple action instances, which can be attributed to THUMOS-14’s higher action density (averaging 15.5 action instances per video compared to ActivityNet’s 1.65 instances per video) and more challenging temporal boundaries with shorter average action durations (9.6 seconds vs. 36.2 seconds in ActivityNet).

3. **MS³L Framework for Action Quality Assessment:** Completing our multi-level framework, we introduced a novel Mamba-based approach for evaluating how well actions are performed—the most sophisticated dimension of temporal understanding. Our key innovations include replacement of computationally expensive attention mechanisms with selective state space models, a Cross-Modal Selection Mechanism for dynamic integration of multiple modalities, and a Temporal Granularity Adaptation module capturing action-specific dynamics at multiple scales. The TGA module consists of parallel Mamba blocks with different temporal scales (expansion factors {1,2,4}), enabling simultaneous capture of both fine-grained movements and global compositional structures. A gating mechanism determines the optimal combination of these multi-scale features based on action characteristics, with final quality prediction achieved through regression on adaptively combined features. Experimental results demonstrated improvements over previous state-of-the-art methods, particularly for complex routines requiring multi-scale temporal understanding, while leveraging insights from both previous levels of our framework.

Together, these contributions establish a coherent framework for multi-level temporal understanding in videos. Our work demonstrates a natural progression from recognizing what actions occur to determining when they happen and finally to assessing how well they are performed. This progression reflects the increasingly sophisticated demands of real-world video analysis applications while maintaining scientific and methodological consistency across all three levels.

The common methodological threads connecting our contributions include hierarchical feature learning, explicit modeling of temporal relationships at various scales, and effective multi-modal integration strategies. These consistent approaches across different levels of our framework highlight the unified nature of our research agenda and its contributions to the broader field of video understanding.

6.3 Limitations and Lessons Learned

While our framework demonstrates measurable improvements across all three levels of temporal understanding, several limitations and challenges warrant honest discussion:

Computational Resource Requirements (Chapter 3-5): The multi-level nature of our framework, while achieving zero additional inference cost for the discriminative learning component in Chapter 3, does introduce computational overhead in the comprehensive

temporal localization (Chapter 4) and multi-scale quality assessment stages (Chapter 5) compared to simpler single-level approaches. This trade-off between performance and efficiency requires careful consideration in deployment scenarios.

Multi-Modal Integration Timing (Chapter 4 & 5): Our current approaches to multi-modal fusion in both the temporal localization system (Chapter 4) and action quality assessment framework (Chapter 5) primarily integrate modalities at relatively late stages in the processing pipeline. While effective, this design choice may miss opportunities for earlier, deeper integration that could reveal complementary information patterns and cross-modal correlations from the beginning of the processing pipeline.

Dataset-Dependent Performance Variations (Chapter 4): Our analysis of the temporal action localization results in Chapter 4 revealed that performance improvements varied significantly across different datasets. The more pronounced improvements on THUMOS-14 compared to ActivityNet v1.3, particularly for videos with multiple action instances, highlighted the relationship between dataset complexity and method effectiveness. This suggests that while our approach handles complex scenarios well, simpler datasets may not fully showcase the method’s capabilities.

Temporal Scale Sensitivity (Chapter 4): Our investigation in the temporal localization framework (Chapter 4) revealed that very long action segments, particularly when evaluated at high IoU thresholds (≥ 0.95), sometimes benefited less from our attention mechanisms. This indicates that our approach is particularly effective for detecting shorter action segments with precise boundaries, but may require additional considerations for extended action sequences with varied internal dynamics.

These limitations provide valuable insights for future research directions and highlight areas where continued investigation could yield additional improvements.

6.4 Future Work

While our multi-level temporal understanding framework has made significant contributions to video analysis, several promising directions remain for future research. These opportunities build upon our work across the three levels of temporal understanding—from action recognition to localization to quality assessment—and extend into emerging areas of artificial intelligence.

6.4.1 Technical Advancements in Video Understanding

The principles of coarse-to-fine-to-finer discriminative learning developed in Chapter 3 show potential for broader application beyond action recognition. Future research could adapt this hierarchical approach to enhance performance in video captioning, where progressively finer distinctions between visual content could generate more precise descriptions; in video question answering, where nuanced discrimination between relevant and irrelevant temporal segments could improve answer accuracy; and in video retrieval, where multi-level discriminative features could enable more accurate matching between queries and video content.

Our current approaches to multi-modal fusion, particularly in Chapters 4 and 5, primarily integrate modalities at relatively late stages in the processing pipeline. A promising direction involves exploring earlier and deeper integration of visual, motion, and audio modalities. This could include co-attention mechanisms between modalities at the feature extraction stage or shared representation learning that captures cross-modal correlations from the beginning of the processing pipeline. Such approaches might reveal complementary information that is lost when modalities are processed independently before fusion.

Another important area for advancement is developing more interpretable video understanding models. While our visualization analyses in Chapter 3 demonstrated that our approach guides models to focus on action-relevant regions, expanding these capabilities to explain temporal dependencies and quality assessments remains challenging. Future work could develop attention visualization techniques specific to temporal models and create methods for highlighting the most influential features in quality assessment decisions. These interpretability mechanisms would be particularly valuable for applications in domains such as healthcare and sports coaching, where understanding why a system made a particular assessment is as important as the assessment itself.

6.4.2 Scaling and Efficiency Considerations for Future Video Understanding

Future video understanding systems will increasingly need to handle longer-form content, higher resolutions, and continuous streams while maintaining computational efficiency. Several promising research directions could address these scaling challenges:

Developing new architectures with sub-linear computational scaling properties will be crucial as video durations extend from minutes to hours or days. While current state space models offer linear complexity compared to quadratic attention mechanisms, further

innovations might achieve logarithmic complexity through hierarchical temporal modeling or sparse activation techniques.

Adaptive computation frameworks represent another promising direction, where models could dynamically allocate computational resources based on content complexity. During periods of low activity or minimal change, processing could be reduced, while semantically rich segments would receive greater computational attention. This approach could significantly improve efficiency for long-duration surveillance, livestreams, or continuous monitoring applications.

Memory-augmented architectures could enhance efficiency by storing and retrieving relevant information rather than repeatedly processing similar content. For instance, in sports analysis or healthcare monitoring, common patterns could be recognized and stored, allowing the system to focus computational resources on novel or irregular events that deviate from established patterns.

Hardware-specific optimization will also be crucial for practical deployment. Future research should explore specialized processing units optimized for temporal operations, distributed computing approaches for parallel video processing, and edge-cloud collaborative architectures that balance local processing needs with more computationally intensive operations in the cloud.

6.4.3 Integration with Large Language Models and Video-Language Models

It is important to note that the research presented in this thesis was conducted prior to the emergence of large language models (LLMs) and video-language models that have recently transformed the AI landscape. At the time of our work, the field of video understanding primarily relied on specialized architectures designed exclusively for visual data. With this historical context in mind, there are exciting opportunities to integrate our work with these emerging paradigms:

Our discriminative learning framework and multi-modal temporal understanding approaches could potentially enhance recent large language models by helping them better distinguish nuanced action concepts when processing video content. This could address the semantic gap that often exists between visual understanding and language capabilities. Similarly, our temporal modeling approaches and state space mechanisms could be incorporated into Video-LLM architectures to enhance their temporal understanding capabilities, particularly for complex action assessment and fine-grained temporal localization tasks that remain challenging even for recent multimodal models.

The techniques we developed for action discrimination and quality assessment could also inform the development of more effective prompting strategies for video-language models. Our multi-level discrimination approach could be translated into prompting frameworks that guide Video-LLMs to focus on progressively finer action details and quality attributes.

In conclusion, this thesis has advanced the state of video understanding through novel approaches to action recognition, temporal localization, and quality assessment. By addressing fundamental challenges in feature representation, temporal modeling, and multi-modal integration, our work contributes to more accurate, efficient, and nuanced video analysis systems. The future directions outlined above present exciting opportunities to build upon these foundations and further push the boundaries of video understanding in an era increasingly defined by multimodal AI capabilities.

6.5 Broader Impact and Final Thoughts

The advancements presented in this thesis have implications that extend beyond technical contributions to the field of computer vision. As our multi-level temporal understanding framework finds applications in various domains, it is important to consider its broader impact on society and its place within the evolving landscape of artificial intelligence research.

Our work on action recognition through coarse-to-fine-to-finer discriminative learning addresses fundamental challenges in video representation, establishing robust foundations for distinguishing what actions occur. These capabilities could enhance applications in content categorization, human-computer interaction, and behavioral analysis, where accurate action recognition enables more intuitive and responsive systems. The model-agnostic nature of our approach makes these improvements widely applicable across different architectural paradigms.

The temporal localization capabilities developed in the second level of our framework extend these benefits to applications requiring precise temporal understanding. By determining exactly when actions occur within continuous video streams, these techniques enable more sophisticated content analysis, event detection, and temporal segmentation. The multi-modal integration strategies we developed for this level demonstrate how complementary information sources can be effectively combined to enhance temporal precision.

Our research on action quality assessment completes the framework by addressing how well actions are performed—a dimension with particular relevance in domains such as sports analysis, skill development, and rehabilitation monitoring. The capacity to evaluate

performance quality opens new possibilities for automated feedback systems, performance optimization, and objective assessment in contexts where quality metrics were previously reliant on human judgment.

Throughout this research, we have maintained a consistent focus on developing a coherent, progressive understanding of temporal dynamics in videos. The natural progression from recognition to localization to quality assessment reflects the increasing sophistication of human visual cognition and establishes a framework that aligns computational approaches with human cognitive processes. This alignment is particularly valuable as AI systems increasingly serve as cognitive assistants and analytical tools across diverse application domains.

It is worth noting that the research presented in this thesis was conducted during a transformative period in AI, preceding the widespread emergence of large language models and multimodal systems that have since reshaped the landscape. Our focus on discriminative feature learning, multi-scale temporal modeling, and multi-modal integration anticipated many of the challenges that remain relevant in today’s evolving video understanding paradigms. The principles established through our framework continue to inform current research directions, even as specific architectural implementations evolve.

As we look to the future, the field of video understanding stands at an exciting crossroads. The integration of state space models, sophisticated temporal modeling, and multi-level feature learning within our framework offers new paradigms for how machines can interpret and analyze visual content across the spectrum from recognition to quality assessment. The continued advancement of these techniques, now increasingly in concert with language understanding capabilities, promises to further bridge the gap between computational systems and human-level understanding of visual information.

This thesis represents a step forward in this ongoing journey, contributing a unified framework for temporal understanding in videos. By systematically addressing the questions of what actions occur, when they happen, and how well they are performed, we have established an integrated approach to video analysis that mirrors the progressive nature of human visual comprehension while advancing the technical capabilities of computational systems. We hope these contributions will inspire continued research and innovation in the fascinating and rapidly evolving field of video understanding.

References

- [1] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *arXiv preprint arXiv:2006.16228*, 2020.
- [2] Humam Alwassel, Silvio Giancola, and Bernard Ghanem. Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. *arXiv preprint arXiv:2011.11479*, 2020.
- [3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021.
- [4] Taivanbat Badamdorj, Mrigank Rochan, Yang Wang, and Li Cheng. Joint visual and audio learning for video highlight detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8127–8137, 2021.
- [5] Yancheng Bai, Huijuan Xu, Kate Saenko, and Bernard Ghanem. Contextual multi-scale region convolutional 3d network for activity detection. *arXiv preprint arXiv:1801.09184*, 2018.
- [6] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021.
- [7] Shyamal Buch, Victor Escorcia, Bernard Ghanem, Li Fei-Fei, and Juan Carlos Niebles. End-to-end, single-stream temporal action detection in untrimmed videos. 2019.
- [8] Jean Burgess and Joshua Green. *YouTube: Online video and participatory culture*. John Wiley & Sons, 2018.

- [9] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–970, 2015.
- [10] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018.
- [11] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019.
- [12] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.
- [13] Manuel Castells. *The rise of the network society*. John wiley & sons, 2011.
- [14] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018.
- [15] Charmve. Computer vision practical exercises: Algorithms and applicationscomputer vision in action. <https://github.com/Charmve/computer-vision-in-action>, 2021.06.
- [16] Soumyabrata Chaudhuri and Saumik Bhattacharya. Simba: Mamba augmented u-shiftgc for skeletal action recognition in videos. *arXiv preprint arXiv:2404.07645*, 2024.
- [17] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.
- [18] Cisco Systems, Inc. Cisco annual internet report (2018–2023). Technical report, Cisco, 2020.
- [19] MMAAction2 Contributors. Openmmlab’s next generation video understanding toolbox and benchmark. <https://github.com/open-mmlab/mmaaction2>, 2020.

- [20] Xiyang Dai, Bharat Singh, Guyue Zhang, Larry S Davis, and Yan Qiu Chen. Temporal context network for activity localization in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5793–5802, 2017.
- [21] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.
- [22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.
- [23] Haodong Duan, Yue Zhao, Yuanjun Xiong, Wentao Liu, and Dahua Lin. Omni-sourced webly-supervised learning for video recognition. In *European Conference on Computer Vision*, pages 670–688. Springer, 2020.
- [24] Stefan Elfving, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning, 2017.
- [25] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *arXiv preprint arXiv:2104.11227*, 2021.
- [26] Gunnar Farneback. Two-frame motion estimation based on polynomial expansion. In *Image Analysis: 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29–July 2, 2003 Proceedings 13*, pages 363–370. Springer, 2003.
- [27] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020.
- [28] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6202–6211, 2019.
- [29] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

- [30] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Cascaded boundary regression for temporal action detection. *arXiv preprint arXiv:1705.01180*, 2017.
- [31] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021.
- [32] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The” something something” video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017.
- [33] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.
- [34] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *arXiv preprint arXiv:2010.09709*, 2020.
- [35] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.
- [36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [37] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *2017 IEEE international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE, 2017.
- [38] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
- [39] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

- [40] Wenbo Huang, Jinghui Zhang, Guang Li, Lei Zhang, Shuoyuan Wang, Fang Dong, Jiahui Jin, Takahiro Ogawa, and Miki Haseyama. Manta: Enhancing mamba for few-shot action recognition of long sub-sequence. *arXiv preprint arXiv:2412.07481*, 2024.
- [41] Wenyong Huang, Wenchao Hu, Yu Ting Yeung, and Xiao Chen. Conv-transformer transducer: Low latency, low frame rate, streamable end-to-end speech recognition. *arXiv preprint arXiv:2008.05750*, 2020.
- [42] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos “in the wild”. *Computer Vision and Image Understanding*, 155:1–23, 2017.
- [43] International Telecommunication Union. Measuring digital development: Facts and figures 2024. Technical report, International Telecommunication Union, 2024.
- [44] Xiao He is not a box. Do not ”blink” challenge, 10 seconds can not see the changes in the picture, not many people can pass!! <https://www.bilibili.com/video/BV1eN4y137nC/>, June 2022. Bilibili Video.
- [45] Henry Jenkins. Convergence culture: Where old and new media collide. *New York, NY: New*, pages 1–2, 2006.
- [46] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.
- [47] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.
- [48] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.
- [49] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

- [50] Dan Kondratyuk, Liangzhe Yuan, Yandong Li, Li Zhang, Mingxing Tan, Matthew Brown, and Boqing Gong. Movinets: Mobile video networks for efficient video recognition. *arXiv preprint arXiv:2103.11511*, 2021.
- [51] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563, 2011.
- [52] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.
- [53] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. In *European Conference on Computer Vision*, pages 237–255. Springer, 2025.
- [54] Yaoxin Li, Deepak Sridhar, Hanwen Liang, and Alexander Wong. Spot the difference! temporal coarse to fine to finer difference spotting for action recognition in videos. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024.
- [55] Hanwen Liang, Niamul Quader, Zhixiang Chi, Lizhe Chen, Peng Dai, Juwei Lu, and Yang Wang. Self-supervised spatiotemporal representation learning by exploiting video continuity. *arXiv preprint arXiv:2112.05883*, 2021.
- [56] Chuming Lin, Jian Li, Yabiao Wang, Ying Tai, Donghao Luo, Zhipeng Cui, Chengjie Wang, Jilin Li, Feiyue Huang, and Rongrong Ji. Fast learning of temporal action proposal via dense boundary generator. In *Proceedings of the Thirty-Fourth Conference of the Association for the Advancement of Artificial Intelligence (AAAI-20)*, pages 11499–11506, New York, NY, USA, 2020. AAAI Press.
- [57] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7083–7093, 2019.
- [58] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation, 2019.
- [59] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 988–996, 2017.

- [60] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.
- [61] Hengshuai Liu, Jianjun Li, Jiale Tong, Guang Li, Qian Wang, and Ming Zhang. Real-time spatiotemporal action localization algorithm using improved cnns architecture. *Scientific Reports*, 14(1):24772, 2024.
- [62] Qinying Liu and Zilei Wang. Progressive boundary refinement network for temporal action detection. In *AAAI*, pages 11612–11619, 2020.
- [63] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3042–3051, 2022.
- [64] Yi Liu, Limin Wang, Yali Wang, Xiao Ma, and Yu Qiao. Fineaction: A fine-grained video dataset for temporal action localization. *IEEE transactions on image processing*, 31:6937–6950, 2022.
- [65] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang. Multi-granularity generator for temporal action proposal. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3604–3613, 2019.
- [66] Yuliang Liu, Chunhua Shen, Lianwen Jin, Tong He, Peng Chen, Chongyu Liu, and Hao Chen. Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.
- [67] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.
- [68] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3202–3211, 2022.
- [69] Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. Tam: Temporal adaptive module for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13708–13718, 2021.

- [70] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 344–353, 2019.
- [71] Lev Manovich. The language of new media, 2002.
- [72] Richard E Mayer. Multimedia instruction. In *Handbook of research on educational communications and technology*, pages 385–399. Springer, 2013.
- [73] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9879–9889, 2020.
- [74] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019.
- [75] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11205–11214, 2021.
- [76] Jongho Park, Jaeseung Park, Zheyang Xiong, Nayoung Lee, Jaewoong Cho, Samet Oymak, Kangwook Lee, and Dimitris Papailiopoulos. Can mamba learn how to learn? a comparative study on in-context learning tasks. *arXiv preprint arXiv:2402.04248*, 2024.
- [77] Paritosh Parmar and Brendan Tran Morris. Learning to score olympic events. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 20–28, 2017.
- [78] Hamed Pirsiavash, Carl Vondrick, and Antonio Torralba. Assessing the quality of actions. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part VI 13*, pages 556–571. Springer, 2014.
- [79] Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. *arXiv preprint arXiv:2103.13141*, 2021.

- [80] Haonan Qiu, Yingbin Zheng, Hao Ye, Yao Lu, Feng Wang, and Liang He. Precise temporal action localization by evolving temporal proposals. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 388–396, 2018.
- [81] Niamul Quader, Md Mafijul Islam Bhuiyan, Juwei Lu, Peng Dai, and Wei Li. Weight excitation: Built-in attention mechanisms in convolutional neural networks. In *European Conference on Computer Vision*, pages 87–103. Springer, 2020.
- [82] Niamul Quader, Juwei Lu, Peng Dai, and Wei Li. Towards efficient coarse-to-fine networks for action and gesture recognition. In *European Conference on Computer Vision*, pages 35–51. Springer, 2020.
- [83] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.
- [84] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [85] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5734–5743, 2017.
- [86] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer, 2016.
- [87] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.
- [88] Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1349–1380, 2000.

- [89] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.
- [90] Deepak Sridhar, Niamul Quader, Srikanth Muralidharan, Yaoxin Li, Peng Dai, and Juwei Lu. Class semantics-based attention for action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13739–13748, 2021.
- [91] Lang Su, Chuqing Hu, Guofa Li, and Dongpu Cao. Msaf: Multimodal split attention fusion. *arXiv preprint arXiv:2012.07175*, 2020.
- [92] Zhi Tian, Hao Chen, Xinlong Wang, Yuliang Liu, and Chunhua Shen. AdelaiDet: A toolbox for instance-level recognition tasks. <https://git.io/adelaidet>, 2019.
- [93] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.
- [94] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5552–5561, 2019.
- [95] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.
- [96] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [97] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60–79, 2013.
- [98] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013.

- [99] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4305–4314, 2015.
- [100] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. *CVPR*, 2017.
- [101] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018.
- [102] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021.
- [103] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.
- [104] Zhengwei Wang, Qi She, and Aljosa Smolic. Action-net: Multipath excitation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13214–13223, 2021.
- [105] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.
- [106] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 284–293, 2019.
- [107] Wenhao Wu et al. Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.
- [108] Wenhao Wu, Zhun Sun, and Wanli Ouyang. Revisiting classifier: Transferring vision-language models for video recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 2847–2855, 2023.
- [109] Wenhao Wu, Xiaohan Wang, Haipeng Luo, Jingdong Wang, Yi Yang, and Wanli Ouyang. Bidirectional cross-modal knowledge exploration for video recognition with

- pre-trained vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6620–6630, 2023.
- [110] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.
 - [111] Boyang Xia, Zhihao Wang, Wenhao Wu, Haoran Wang, and Jungong Han. Temporal saliency query network for efficient video recognition. In *European Conference on Computer Vision*, pages 741–759. Springer, 2022.
 - [112] Boyang Xia, Wenhao Wu, Haoran Wang, Rui Su, Dongliang He, Haosen Yang, Xiaoran Fan, and Wanli Ouyang. Nsnet: Non-saliency suppression sampler for efficient video recognition. In *European Conference on Computer Vision*, pages 705–723. Springer, 2022.
 - [113] Jinglin Xu, Yongming Rao, Xumin Yu, Guangyi Chen, Jie Zhou, and Jiwen Lu. Finediving: A fine-grained dataset for procedure-aware action quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2949–2958, 2022.
 - [114] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10156–10165, 2020.
 - [115] Yan Yan, Chenliang Xu, Dawen Cai, and Jason J Corso. Weakly supervised actor-action segmentation via robust multi-task ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1298–1307, 2017.
 - [116] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal pyramid network for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 591–600, 2020.
 - [117] Ling-An Zeng, Fa-Ting Hong, Wei-Shi Zheng, Qi-Zhi Yu, Wei Zeng, Yao-Wei Wang, and Jian-Huang Lai. Hybrid dynamic-static context-aware attention network for action assessment in long videos. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2526–2534, 2020.
 - [118] Ling-An Zeng and Wei-Shi Zheng. Multimodal action quality assessment. *IEEE Transactions on Image Processing*, 2024.

- [119] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7094–7103, 2019.
- [120] Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. Hacs: Human action clips and segments dataset for recognition and temporal localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8668–8678, 2019.
- [121] Jiaojiao Zhao, Yanyi Zhang, Xinyu Li, Hao Chen, Bing Shuai, Mingze Xu, Chunhui Liu, Kaustav Kundu, Yuanjun Xiong, Davide Modolo, et al. Tuber: Tubelet transformer for video action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13598–13607, 2022.
- [122] Y Zhao, B Zhang, Z Wu, S Yang, L Zhou, S Yan, L Wang, Y Xiong, D Lin, Y Qiao, et al. Cuhk & ethz & siat submission to activitynet challenge 2017. *arXiv preprint arXiv:1710.08011*, 8, 2017.
- [123] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2914–2923, 2017.