Ensemble Learning with Attention and Audio for Robust Video Classification

By

Dereje Tadesse Abzaw

Submitted to the School of Information Technology and Engineering

in partial fulfillment of the requirements for the degree of

Master of Science in Artificial Intelligence

College of Technology and Built Environment

Addis Ababa University

Addis Ababa, Ethiopia

June 2025

Approval

This is to certify that this thesis titled ***Ensemble Learning with Attention and Audio for Robust Video Classification*** is prepared by ***Dereje Tadesse Abzaw*** and submitted in partial fulfillment of the thesis-option requirements for the Degree of Master of Science in Artificial Intelligence at School of Information Technology & Engineering, Addis Ababa Institute of Technology.

Advisor

Dr. Beakal Gizachew

Name	Signature	Date

External Examiner

Dr. Mesfin Abebe

Name	Signature	Date

Internal Examiner

Dr. Fantahun Bogale

Name	Signature	Date

Chairperson

Dr. Henok Mulugeta

Name	Signature	Date

Abstract

The classification of video scenes is a fundamental task for many applications, such as content recommendation, indexing, and monitoring broadcasts. Current methods often depend on annotation-dependent object detection models, restricting their generalizability when working with different types of broadcast content, particularly cases where visual clues like logos or brands may not have clear definition or presence. This thesis is intended to address the problems associated with current methods through describing a two-stage classification framework that integrates both recognized and unheard information to improve accuracy and robustness of classification. The first stage utilizes a detection model based on pretrained models of object detection and enhanced spatial attention to detect physical visual markers (such as program logo or branded intro sequences) in video program content. However, individual visual indicators are sometimes not robust enough to add confidence, especially in content such as situational comedies where logos do not exist. The second stage describes a two-staged, early fusion ensemble presentation of convolutional neural network-based visual features and recurrent neural network-based audio features. The two modes each use some complementary properties, thus could be used for more robust classification. Experiments were completed with a dataset of approximately 19 hours of content from 13 TV programs across three channels, all focused on intro, credit, and outro segments. The visual-only model achieved 96.83% accuracy, while the audio-only model achieved 90.91%. The proposed early fusion ensemble method achieved 94.13% accuracy and revealed more robustness in difficult situations when quality of visual data was low or ambiguous. Ablation studies contrasting model performance with different ensemble methods confirmed the greater utility of early fusion and its capturing of cross-modal interactions. The system is also designed to be computationally efficient allowing for operationalization in broadcast media settings. This work, while also demonstrating methodical video classification ability, fills a significant gap for scalable and generalizable video classification through the integration of multimodal learning, especially with large amounts of uncontrollable annotations which has previously been a hurdle to more typical models.

Acknowledgements

I would like to express my deepest gratitude to my parents, **Tadesse Abzaw** (Babi) and **Genet Aweke** (Genaye) for their continuous support, belief and faith in me. Their love and encouragement helped me to reach this finish line.

I would also like to thank and give gratitude to my advisor, **Dr. Beakal Gizachew** for his advising and mentorship. I cannot say how valuable you have been to me in this process and I owe you big time. I would also like to thank **Dr. Elfelious Getachew**, **Dr. Adane Mamuye**, **Dr. Fantahun Bogale**, **Dr. Henok Mulugeta**, **Dr. Sileshi Demesie**, and **Dr. Sammy Assefa** for all of the feedback I receive from you.

My deepest gratitude goes to **Dr. Ben Geortzel** and **Dr. Natnael Argaw**; thank you for believing in me and giving me the opportunity to work at iCog Labs. That opportunity has been of incredible value to me and has been critical for the person I am today, and I will never take your support for granted.

I would like to take this opportunity to thank my friend **Tewdros Wondifraw** for his encouragement and support; he suggested that I work on this thesis theme, and I am grateful for his boldness, confidence, and belief in me. Also, I would like to thank my friend **Amanuel Negash** for being the reason I joined the program. He thinks highly of me. A very dedicated fellow who inspires you to work hard and commit, day in and day out, to reach his level. He helped me a lot. It would be a big mistake not to mention **Micheal Sheleme** and **Selameab Setargew** for helping me finish this Thesis. I appreciate your guidance, advice, and mentorship through the whole process.

I would also like to give my utmost thanks to **Dr. Aemro Tadesse** (MD), **Kalkidan Tadesse**, **Abrham Amare**, **Amanuel Digafe**, and **Beamlak Teshome**. Thank you for your love and support

Dedication

To my precious niece,

Mela Misiker,

who left this world so soon. Your time with us was short, but the joy and love you brought us will linger forever. We are heartbroken beyond words. Every day, we feel the heartache of your absence. To your parents, Kal and Miskiru, whose love and strength will forever be a beautiful tribute to your brief life, this dedication is for them, as well. While we mourn you, we also find comfort and hope in knowing we will see you again one day. You are immensely missed, thoroughly loved and will aways be remembered.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AP Average Precision.

ASR Audio Speech Recognition.

BMCC Broadcast Media Content Classification.

BoAW Bag of Audio Words.

C3D Convolutional 3D Networks.

CNN Convolutional Neural Network.

ConvFeats Convolutional Features.

ConvLSTM Convolutional-Long Short-Term Memory.

DCGN Deep Convolutional Graph Network.

DNN Deep Neural Network.

GAN Generative Adversarial Networks.

GMM Gaussian mixture models.

HMM Hidden Markov Model.

I3D Inflated 3D Convolutional Networks.

LSTM Long Short-Term Memory.

mAP Mean Average Precision.

MFCC Mel Frequency Cepestral Coefficients.

mIoU Mean Intersection Over Union.

MRR Mean Reciprocal Rank.

RNN Recurrent Neural Network.

SIFT Scale Invariant Feature Transform.

SVM Support Vector Machines.

VC Video Classification.

VidSGG Video Scene Graph Generation.

VOD Video Object Detecetion.

VOS Video Object Segmentation.

VSS Video Scene Segmentation.

Chapter 1 Introduction

Video content has become our integral part of our daily lives, whether it is on YouTube, TikTok, Twitter, or virtually any social media platform, playing a role in everyday activities[1]. As a result, one of the fundamental tasks in computer vision has become video classification task [1, 2]. It is as simple as assigning a predefined label to a certain video clip and classifying them accordingly, just like image classification [2]. It is important for different applications including content recommendation, surveillance, video indexing and broadcast monitoring [2, 3, 4].

Video classification is computationally expensive in needing many frames to process[5] and extracting motion information [6]. One of the approaches in running models related to video classification has been decomposing the problem into a lesser tasks [5]. The primary challenge in this process, of course, is representing a video in a way that can make an effective video understanding [7]. There has been an approach in representing video with a frame-level and video-level understanding [3, 8]. The current trend in frame-level representation, with different approaches, does involve the selecting of key frames that are able to represent the entire video scene perfectly [3, 9, 10, 11]. Convolutional Neural Networks have been known to play an emaculate role in visual-based video classification [12, 13, 14]. Together with attention-cluster approach have captured a global attention and has been the state of the art approach in classifying videos, as attention compliments CNNs long range dependency and focus on key visual element limitations [14].

In parallel, audio-based classification has also emerged, complementarily, when visuals are ambiguous or occluded [15]. One edge of audio-only classification method is that it is less expensive computationally than visual-based approaches [16]. It is also known to have less size in manipulating and generally working with audio clips [16, 17].

Combining the strength and weakness of both modalities, in a mixture of experts scheme, presents a promising solution. Not only that, but, this ensemble strategy, solves the struggle of VC models to balance accuracy and efficiency when generalizing across varied content types [1, 15, 18] .

1.1 Background

Video Classification (VC) has become an interesting task lately. Applications in computer vision tasks, content recommendation, surveillance, autonomous systems, to name a few, have made them very popular [19]. One thing that has revolutionized VC tasks lately, as mentioned above, is the availability of online video resources that we go about daily and the advent of deep learning techniques [19, 20].

Attention mechanisms have further advanced the manipulation and handling of videos lately, especially in working with long-range dependencies and complex temporal relationships [21, 22]. However, they come with an increased computational cost, as we are adding an additional layer to the existing VC models [21].

Similar to attention mechanisms, the efficiency issues in VC models have also been addressed using ensemble learning [22]. These models try to balance accuracy and efficiency, making them a good fit for environments with limited resources. In addition to the visual modality, audio, just like text, has proven to be very important [22]. Together, visual and audio cues help fully represent what's happening in a video [22, 23].

This thesis builds on these developments by exploring how attention mechanisms, ensemble strategies, and audio features can be effectively combined to create more accurate and efficient video classification systems.

1.2 Motivation

Videos have rich and diverse data, with different images and scenes situated in different contexts. Typically, a video will not be interpreted consisting of solely the visuals, as we tend to rely on many factors to help our interpretation [22, 24]. Using audio, in much the same way we have relied on text previously, can help the gap of misinterpretation and misrepresentation [15]. It is difficult to tell from a video clip whether a person is laughing, clapping or talking. However, adding audio combined with these visual cues, we can disseminate the content more accurately.

As we develop neural networks that become deeper and more intricate, we need to think about how much more computational power they require and how feasible it is to use them, especially on smaller devices and in communities with limited resources [15, 22, 25]. This brings up the need for solutions that can keep a good balance between performance and efficiency. Smaller models and using a mix of models together seem like a good way forward, since they can work well without making the system much bigger or slower.

Attention mechanisms compliments the model focus on important parts of the inputs [13, 14]. By ignoring things that don't matter, the model understands easier and performs well. The main idea behind this work is the working of attention together with audio and visual cues. The general goal is to build a Video classification system that is accurate and also practical but also provides a real-world solution having a lower computing power and a data quality Attention mechanisms are intended to improve this configuration further by forcing the model to focus only on the most important aspects of both the visual and audio inputs [13, 14]. This both helps in understanding the model's actions and assists the model in performing well through the disregard of unimportant stimuli. One fundamental takeaway from this work is the extent to which attention, light-weight working together models, and audio-plus-visual (AV) data work together. It also hopefully illustrates a path to making video classification systems both accurate and usable (in the sense of performing in real-world scenarios that will often have computational and/or data quality limitations).

Modern video classification models, particularly ones that consider temporal dynamics, are generally based on 3D convolutions, recurrent layers, or transformers [15, 22, 25]. All of these architectures are resource expensive and are also not amenable to real-time applications or devices with limited computational power. The main motivation of this work is to reduce this complexity while maintaining performance as much as possible.

High-capacity models require significant memory bandwidth and storage, both for training and inference [26]. In many use cases—such as mobile applications, edge devices, or in the context of Ethiopian broadcast infrastructure—such hardware resources may be scarce.

Designing memory-efficient solutions that retain strong predictive power is thus an important challenge addressed in this thesis.

Using visual features alone can lead to misclassification [22], particularly when dealing with broadcast content, where similar visual styles may exist across different programs. Audio features provide rich semantic context when used along with visual features. In instances, such as genre analysis, background music, tone of speech, and ambient sound can help separate genres. This thesis utilizes both modalities to improve classification performance.

The attention mechanism allows a model to disregard everything else in the input while focusing on the most informative frames (or audio segments) instead [22]. In doing so, it improves model performance and interpretability. In broadcast video, attention can make a difference by filtering out irrelevant or repeated frame content that may confuse the classifier, as it allows the model to efficiently focus on data most relevant to classification with less noise to deal with in processing.

Finally, the desire is to create systems that are both accurate and usable, specifically scalable for use with large video datasets and run on constrained devices [15]. This is especially important for developing countries and broadcasters that do not have high-performance computing resources.

1.3 Statement of the Problem

Classifying videos is still a difficult task, especially when working with real-world content like broadcast media, where a single film may include a variety of scenes, subjects, or modalities [27, 28, 29, 30, 31, 32]. The majority of current models are either trained for specific content domains or rely solely on visual features, despite the fact that deep learning has greatly advanced the discipline [33, 34]. Their capacity to generalize across a variety of video material formats is thus limited.

Audio, which frequently contains important semantic information that enhances visual signals, is difficult to include into many current techniques [22, 35, 36, 37]. For instance, if the sounds that accompany two visually comparable pictures are different, they may fall into completely separate categories [22, 37]. Unfortunately, the majority of models simply disregard audio or only use it sometimes [35] , which results in an inadequate comprehension of the material .

Additionally, processing frame by frame for deployment in large-scale or real-time contexts is not always possible and is computationally expensive [38]. While methods such as keyframe selection have been proposed to reduce processing overhead [9, 11, 38], they often overlook aural context and temporal consistency, which are both important for accurate classification [7, 8].

Furthermore, multimodal approaches sometimes depend on annotated subtitles or metadata, which may not be reliable or accessible [1, 12]. This dependence compromises the generalizability and accessibility of current solutions. These limitations, which include modality imbalance, lack of generalizability, and high computational cost, underscore the need for a precise and effective video classification framework that can efficiently leverage both audio and visual information without requiring extensive annotation or domain-specific tuning.

In this thesis we propose an ensemble learning methodology that utilizes audio and attention mechanisms to address these concerns and improve accuracy in video classification tasks. Thus the problem statement is: *How can we create a generalizable video classification model that effectively utilizes audio and visual modalities, without the need for extensive annotations or domain specific training?* Most existing models in the field of video scene segmentation are typically optimized for specific genres or domains such as sports, movies, or news. This specialization limits their generalizability. In contrast, the framework proposed in this thesis is designed to support a wide range of video content types without requiring any domain-specific adjustments, thereby addressing the need for a more adaptable and universally applicable solution. Another significant limitation of many current approaches is the underutilization of auditory information. While some models incorporate minimal auditory cues, they often do not leverage them meaningfully. This research integrates audio and visual modalities through an attention-based fusion strategy, providing a more enriched and complementary source of information for improved classification accuracy. Finally, many existing techniques rely heavily on richly annotated data, such as subtitles or other textual labels. In contrast, the proposed method operates effectively with low-level visual and auditory features, minimizing the need for detailed annotations. This reduces the dependency on labeled datasets and enhances the scalability of the system across diverse video content. The approach is evaluated on datasets collected from broadcast media. Our results show that our approach outperforms state-of-the-art methods for video classification. The following are clear research questions that are poised to be answered in this thesis.

1.4 Research Question

RQ-1: What is the impact of combining audio-visual features through ensemble learning on video classification performance?
RQ-2: To what extent do attention mechanisms improve the accuracy and robustness of video classification across diverse content types?
RQ-3: Can audio cues improve classification outcomes in scenarios with weak or missing visual context?

1.5 Objectives of the Study

1.5.1 General Objective

The main aim of this research is to develop an accurate video classification system by ensembling an attention-based visual model with audio.

1.5.2 Specific Objectives

- To collect a balanced set of videos from different categories like news, sports, and entertainment.
- To develop simple but effective ways to pull important feature from both the video and its sound.

- To utilize attention-based mechanisms to enable the system to attend to the information present in the video that is most pertinent to the task.
- To create an ensemble system using multiple smaller models that use audio and visual cues that will work together.
- To test the system on actual data, check the performance to ensure speed and accuracy and compare with other methods.

1.6 Scope

This research aims to create a video classification system that uses both visual and audio cues, relying on simple but effective models with attention to important features. The focus will be on videos from Ethiopian broadcast channels, including categories like news, sports, and entertainment.

Although the system is developed and tested mainly for broadcast content, it is built to work well with different types of videos with some fine-tuning. Furthermore, although human interpretation of the model's outputs—especially in terms of how meaningful or relevant a classification is—will play a role in evaluation, this will be limited to qualitative insights. The main evaluation will rely on standard performance metrics such as accuracy, efficiency, and generalization across content types. Broader applicability to other domains may be explored if time and resources allow.

1.7 Significance of the Study

Video classification is important in many areas like media tracking, recommending content, security, and searching through large amounts of videos. With so much video being created every day, we need methods that can understand this data accurately but without needing too much computing power.

This study is important because it focuses on building a system that combines both the sound and visuals from videos, using attention techniques to improve how well it works.

This approach is designed to work quickly and dependably on everyday computers, making it practical for real-world use where computing power may be limited

By working with broadcast videos—such as news, sports, and entertainment—the research looks at a variety of content types that many people watch every day. This study's results may pave the way for better tools that automatically sort and understand videos, making it simpler for businesses and researchers to work with large amounts of video data. The ideas developed here could also encourage new approaches to video classification that use multiple sources of information in an efficient way.

1.8 Organization of the Thesis

This thesis begins with an introduction to the major components by identifying the problem, motivation, goals, and limitations of the project to develop an attention-based video classification algorithm based audio and visual features.

The second chapter outlines relevant related work associated with video classification, existing multimodal feature extraction models and research, attention mechanism models and research, and research described ensemble learning methods. This previous work sets context for the proposed method.

The third chapter describes the methods of this study, including how the dataset was collected and prepared, how the audio and visual feature extraction modules were developed, and how the ensemble model was developed and trained.

In Chapter Four, the experimental results are presented and the performance of the proposed system is evaluated regarding accuracy, efficiency and applicability.

In Chapter Five, the conclusions of the thesis will be made, summarizing the findings and addressing their contribution, and providing future directions in efficient multimodal video classification.

1.9 Contribution

A video classification system with the application of an attention mechanism for effectively integrating audio and visual features is presented in this thesis. The system introduces a robust ensemble learning approach that not only enhances classification accuracy but also reduces computational complexity, making it suitable for deployment in resource-constrained environments. In contrast to conventional single-modality models, the proposed framework fuses both audio and visual modalities while taking advantage of their complementary strengths by using attention-based fusion. Furthermore, the research will validate the system against practical broadcast media content, which provides a real-world evaluation of the proposed framework in diverse, noisy environments with lighting conditions and camera motion, and with low resolution footage.

Chapter 2 Literature Review

2.1 Foundations

Video classification (VC) is the process of assigning a category to a video or segments based on its content and visual input [39, 40]. It plays a vital role in many applications, including content moderation, recommendation systems, surveillance, and broadcast media. Contrary to image classification, video requires the analysis of image frames over time. The challenge lies in capturing how contents evolve across time. For example, the difference between a news and a sports event often depends on certain objects, like an anchor or a soccer field, and their movements [41]. In addition to frames, videos contain audio signals that complement the visual information provided. This adds a layer to the semantics of the whole video understanding task.

This section provides the groundwork for understanding video classification techniques. The discussion begins with an examination of video structure, followed by a general overview of typical model architectures and the significance of integrating multiple modalities.

2.1.1 Spatiotemporal Data: What Makes Video Unique

Video data moves through space and time, a feature that complicates its analysis. A video contains many pictures, called frames, which show one after another [40]. Viewers consider the parts of each picture plus how they appear over time. This combination of space and time separates video data from other kinds of data [24].

Space features describe the parts of a single picture, such as objects, textures, scenes along with colors . Time information shows how these parts move or change. For instance, to know if a person walks, a system identifies the person's shape also understands their movement between pictures [42, 43].This blend of space and time needs models that pull out useful patterns from single pictures as well as then learn how those patterns change - this needs structures that use more than the standard 2D convolutions from still pictures [44]. They include tools like 3D convolutions, recurrent layers, or attention based systems which follow

connections through time [7].Videos often have sound that matches the pictures. That adds more detail to the data and allows for multimodal learning [44]. Understanding the video then comes from both what a viewer sees plus what a viewer hears.
A video can be represented as a function over discrete time steps:

$$V : \{1, 2, \dots, T\} \to \mathbb{R}^{H \times W \times C} \tag{2.1}$$

$$V(t) = F_t \in \mathbb{R}^{H \times W \times C} \tag{2.2}$$

where:

- T is the total number of frames in the video,
- H and W are the height and width of each frame,
- C is the number of color channels (e.g., 3 for RGB),
- F_t is the frame at time step t.

Thus, the entire video V can be modeled as a 4D tensor:

$$V \in \mathbb{R}^{T \times H \times W \times C} \tag{2.3}$$

This spatiotemporal structure is what distinguishes video from static images; each frame captures spatial information like objects, textures, layout, while the sequence of frames over time provides temporal cues (motion, transitions, and events) [40, 45].
To effectively classify or analyze videos, models must learn from both the spatial features within each frame and the temporal dependencies across frames. This added temporal dimension significantly increases the complexity of video analysis tasks compared to image-based tasks.

2.1.2 Temporal Modeling Approaches

Understanding the time element of video is a central problem for video classification. An image input stands alone, but video frames arrive in a line and often connect to one another. This requires systems that show how scenes, objects along with actions alter with time [8]. Previous studies originating in temporal modeling used RNNs and specifically Long Short-Term Memory (LSTM) networks, as these models conceptualized video as a temporal sequence of features that each frame yielded generating temporal dependent relationships between all the frames in the respective video [46, 47, 48]. However, RNNs have difficulty with lengthy sequences, which occurs because of gradient vanishing and the size of parallelization [49**?**].

To fix these problems, 3D Convolutional Neural Networks (3D CNNs) came into use. Standard 2D CNNs only get spatial details, but 3D CNNs stretch convolutions into the time dimension. That lets them learn spatiotemporal features together. C3D in addition to I3D (Inflated 3D ConvNet) are models that show this method [44, 50]. They work well, but they use a lot of computer power and need much training data.

Transformer-based models and self attention systems began to see use in video analysis. These systems can find long range connections across time - they learn to give frame importance a variable weight [36, 51]. They perform well, particularly when they connect with pre trained vision backbones.

In short, temporal modeling changed from plain sequential models to more complex spatiotemporal designs. Picking the correct method often means finding a middle ground; this includes accuracy, training expense as well as the ability to model involved changes over time.

Temporal modeling focuses on capturing how information changes across time in a video sequence. Unlike static image classification, video classification must model not only the content of individual frames but also their temporal relationships.

Let the input video sequence be represented as:

$$V = \{F_1, F_2, \dots, F_T\}, \quad \text{where } F_t \in \mathbb{R}^{H \times W \times C} \tag{2.4}$$

After extracting spatial features from each frame (e.g., using a CNN), we denote the feature vector at time t as:

$$x_t = \text{CNN}(F_t), \quad x_t \in \mathbb{R}^d \tag{2.5}$$

Temporal models aim to learn a function f that maps the sequence $\{x_1, x_2, \dots, x_T\}$ to a compact representation h_T:

$$h_t = f(x_t, h_{t-1}) \tag{2.6}$$

For instance, in a Recurrent Neural Network (RNN), this function is typically defined as:

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b) \tag{2.7}$$

where:

- h_t is the hidden state at time t,
- W_h and W_x are weight matrices for the hidden state and input, respectively,
- b is a bias term,
- σ is a non-linear activation function such as tanh or ReLU.

After processing the full sequence, the final hidden state h_T can be used for classification:

$$\hat{y} = \text{softmax}(W_o h_T + b_o) \tag{2.8}$$

Temporal modeling can also be performed using Temporal Convolutional Networks (TCNs), Transformers, or 3D CNNs that operate across both spatial and temporal dimensions simultaneously [52, 53].These approaches are essential for capturing motion patterns, action transitions, and scene changes—key factors in understanding dynamic video content.

2.1.3 Fusion Strategies

Simply understanding something in a video requires understanding what is present in each frame (i.e., the spatial features) and how things evolve across frames (i.e., the temporal features), as well as how to meaningfully fuse the two together [40, 42, 43]. How to fuse spatial and temporal features together is referred to as spatiotemporal fusion, and understanding how to do this is critical to building robust video classification models [54]. Typically, there are three levels of fusion that can happen:

2.2 Fusion Strategies for Spatiotemporal Learning

Fusion strategies for combining spatial and temporal features in video understanding can be broadly categorized into early, mid-level, and late fusion. Early fusion combines spatial and temporal features directly at the input level, typically by applying 3D convolutional neural networks (CNNs) to sequences of raw video frames. This allows the model to learn spatiotemporal patterns simultaneously, as illustrated in Figure 1-a. While this method enables tightly coupled modeling of motion and appearance, it requires large amounts of data and high computational power. Mid-level fusion involves extracting spatial features using a 2D CNN and temporal features using a recurrent neural network (RNN) or temporal convolutional layers. These two feature streams are then merged using techniques such as concatenation or attention, as shown in Figure 1-b. This approach offers a modular design that often leads to improved generalization and reduced overfitting, while maintaining moderate computational efficiency. Late fusion, in contrast, involves making independent predictions from the spatial and temporal branches and combining these outputs through simple strategies such as averaging or voting, or through trainable methods. As depicted in Figure 1-c, late fusion is straightforward to implement and scales easily, but often fails to capture fine-grained interactions between spatial and temporal cues. These fusion strategies offer different trade-offs between accuracy, complexity, and resource requirements [55, 56, 57]. Advanced models often use multi-stream architectures, where different streams process spatial frames, motion (e.g., optical flow), or audio, and are fused using attention or gating

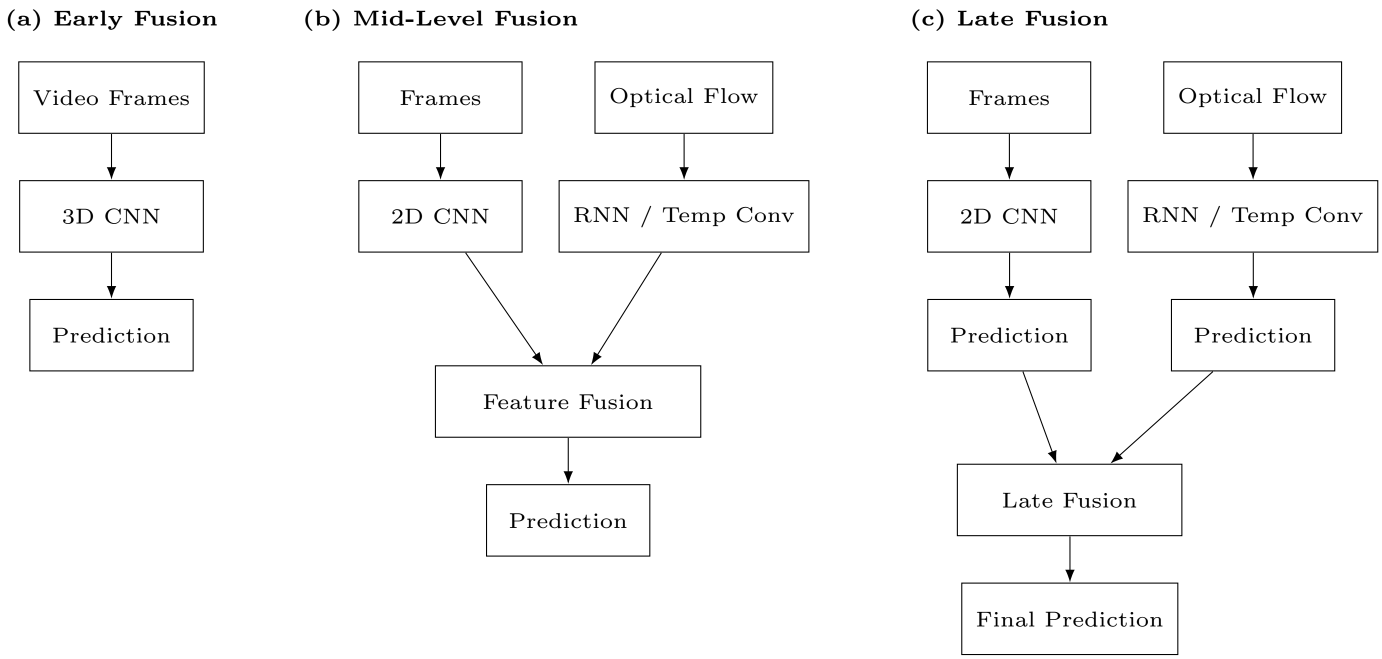


Figure 1: Fusion strategies: (a) Early fusion of raw inputs; (b) Mid-level fusion of visual branches; (c) Late fusion via prediction ensemble.

mechanisms. The choice of fusion strategy impacts not only performance but also the interpretability and efficiency of the model.

2.2.1 The Role of Multimodal Fusion in Enhancing Video Understanding

In many videos in the real world—especially those in a broadcast context—audio and visual channels of information complement one another [37]. While visual frames can describe what is happening, audio provides additional richness with speech, background noise, music, or tone. By combining both audiovisual channels, a model can make more informed decisions and arguably better predictions.

For example, two scenes that may be visually similar—such as an anchor/anchor on a news network and a host delivering sports news from a studio—may look similar based on the background and the way its framed [31, 35]. While both may look similar, their audio tracks will differ: ”news” broadcasts may contain formalized language and low-key, background music, while ”sports” segments may contain hyperbole and background noise from a roar of a crowd [27, 35].

There are a number of reasons for incorporating audio with visual information. First, audio can help to resolve ambiguity in cases where visual channels are ambiguous, making it an important disambiguation tool. Second, audio aids contextual augmentation, helping to establish mood or context for the viewer, and aids in recognizing the category of video. Third, offers robustness and redundancy—when one of the channels does not work (blurry visuals or muted audio) the other can provide adequate functionality with. Lastly, multimodal fusion can facilitate efficiency in learning because models that learn audio-visual representations jointly, leverage the temporal structure and narrative in a video with greater fidelity.

Current video classification systems usually utilize dual-stream architectures composed of one branch reading from the visual representation of video frames, and one branch reading from the audio of either spectrograms or waveforms [37, 55]. Fusion is generally achieved through attention or shared embedding space [58].

By engaging both visual and auditory channels, video understanding models are more global in terms of context, making them especially useful in the broadcast domain since it is important to maximize semantic richness and clarity.

2.2.2 Challenges in Processing Video Data

Working with video data has many unique challenges, making it much more difficult to handle than still image data or text data [59]. A major challenge is the sheer size of video data [60]. Clips as short as a few seconds can have hundreds or thousands of frames and contain a lot of visual information. The large file size can be a barrier to both storage and processing [60]. Another challenge is the time dimension. Video isn't merely a collection of frames or images, because video has the dynamic quality of how scenes and activities change over time [52, 53]. In order to identify meaningful patterns across frames everything needs to recognize continuity and context, which typically requires more innovative architectures and sophisticated training protocols compared to image-based models. Variations in video are also problematic. As illustrated in Figure 2, different lighting, camera angles, motion blur, and occlusions can change the clarity or consistency of frames. In the broadcast space, for example, before and after transitions from one scene to another can be jarring (e.g., going from an anchor to a reporter on a live program, or a commercial break) [27, 31, 35]. As well as transfer learning, sudden cinematic shifts can add confusion when it comes to segmenting content into specific units and understand how to sequence and flow that content [61].

Moreover, alignment between audio and vision is not perfect [35]. For example, when there is a visual scene change, but the background sound or the verbal turn does not change at the same time, it can confuse models that seek to successfully fuse both input streams with sufficient association an exact relationship [18]. To synchronize audio-visual information, both input streams require careful design and considerations against any loss of information or misinterpretations [10].

Finally, the annotation and labeling of video datasets that are long-form is a bottleneck [62]. The manual annotation of long form video datasets has a time cost, which is prohibitive and often expensive [62]. As a result, a lot of researchers have turned to semi-supervised, weakly supervised, or unsupervised approaches to lower the reliance of large labeled datasets [4, 63, 64, 65, 66, 67].

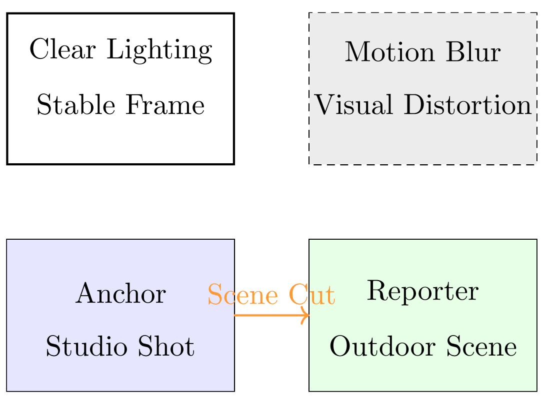


Figure 2: Examples of video variability and scene transitions.

2.2.3 Ensemble and Attention Paradigms in Deep Video Learning

Most video understanding tasks have relied largely on supervised learning approaches like those using Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) [12, 19, 62, 68]. CNNs have been used effectively for spatial feature extraction from each frame of a video while RNNs, especially Long Short Term Memory (LSTM) networks, have been employed for modeling temporal dependencies from a sequence of frames [69]. When combined, these architectures have produced satisfactory results in video classification and segmentation [12, 69]. However, supervised learning comes with several drawbacks, especially reliance on vast amounts of labeled datasets. The annotation process to create these datasets can be costly, labor intensive, and not practical in the vast, varied, and unbounded length of all video content (e.g., broadcast media, etc.).

To overcome these limitations, a number of other learning paradigms, often called semi-supervised, unsupervised or self-supervised learning, have gained traction [62, 63, 64, 70, 71]. Many of the recent self-supervised learning methods (e.g., contrastive learning [72]) are focused on understanding the similarities and differences between one video (or video segment) and another without requiring a large number of labels to characterize the viewing experience [62]. While many of these paradigms are promising, there are at least two different ways that may enhance performance and generalizability while also being largely ignored: ensemble learning and attention.

In this section, we survey current developments that apply ideas from ensemble and attention paradigms to deep video learning, with a focus on the improvements of model robustness, efficiency and temporal-spatial reasoning ability in the context of primitives associated with a complicated video scenario.

2.2.4 Ensemble Learning in Deep Video Models

Combining models through the process of ensemble learning refers to the process of combining several models to outperform a learning model itself [73]. It reduces the individual weaknesses of a model by taking advantage of all the strengths of the models being combined and often results in a model that has better generalization performance, robustness, and stability—all of which are important aspects for video classification and segmentation problems which are highly variable within the class itself and highly complicated over different time series [19, 74, 75]. Here are Ensemble strategies that are commonly known:

2.2.4.1 Bagging

Bagging (Bootstrap Aggregating) is an important method of ensemble learning that generates better predictive performance by reducing variance through model aggregation [18, 76]. Bagging reduces variance by developing multiple training data subsamples through random sampling, usually with replacement, and training independent base models learning from each subsample [76]. Predictions on new data are then formed using aggregated outputs from the models, generally, a simple majority voting for classification tasks and averaging for regression tasks [18]. (see Figure 3).

In deep video learning, bagging combines multi-models on complex datasets from videos, stabilizing the model performance by training independence and promoting diversity across models who vary only slightly in their views of the same data [77]. This problem is prevalent in high-dimensional video with features that accommodate high-risk factors associated with overfitting found using a single model [78]. Bagging model aggregation methods have been more specific to Random Forest, where the technique was initially applied to decision trees has inspired similar adaptations of the method to deep learning, in the parallel training of CNNs or spatiotemporal backbone models on bagged datasets to improve generalization [18, 76]. Bagging assumes data independence, its principles can still enhance deep models by encouraging diversity across network initializations, data augmentations, or even feature subsets [18, 76]. These ideas are especially powerful in large-scale video classification tasks, where the sheer volume and variability of content benefit from model ensembles trained on diverse data slices [19].

Bagging builds multiple models independently on different bootstrapped subsets of the training data [76]. The final prediction is typically made by averaging or majority voting [18, 75] as described in Equation 2.9.

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ be the training set, and let M models be trained on M bootstrapped samples $\mathcal{D}_1, \ldots, \mathcal{D}_M$. Each model f_m predicts class probabilities $p_m(y|x)$.

The final prediction is computed as (see Equation 2.9):

$$\hat{y} = \arg\max_{y} \left(\frac{1}{M} \sum_{m=1}^{M} p_m(y|x) \right) \tag{2.9}$$

This reduces variance and helps stabilize predictions in high-dimensional video feature spaces.

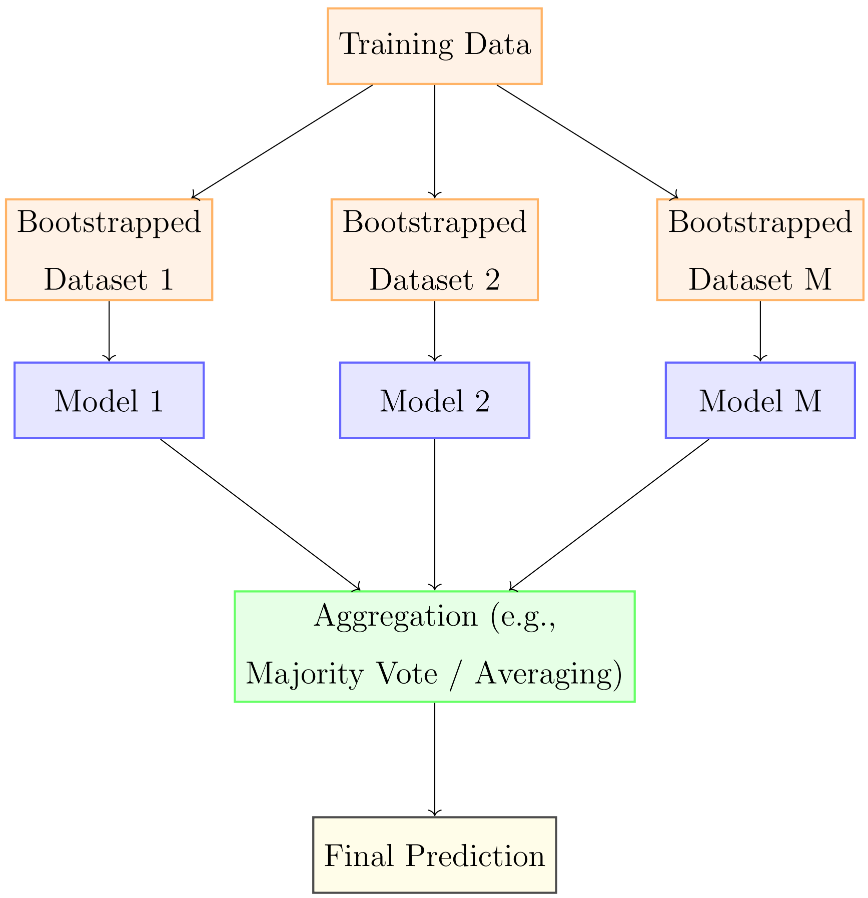


Figure 3: Bagging ensemble with bootstrapped datasets and model aggregation.

2.2.4.2 Boosting

Boosting is another powerful ensemble learning strategy that seeks to elevate weak learners' performance through a sequential process that emphasizes the previous model's mistakes

[18, 79]. Unlike bagging, boosting builds base learners sequentially meaning that the new model learns from previous mistakes [80]. see Figure 4

AdaBoost and Gradient Boosting are classical examples of this approach [81]. For example, AdaBoost, achieves this by assigning higher weights to misclassified examples and combining weak learners by weighted majority voting [80]. Focusing attention on hard-to-classify instances can and will lead to an ensemble's successive improvement in its decision boundaries [82]. Similarly, for regression, a linear combination of weak learners can achieve the same effect [82, 83].For deep video learning, boosting principles are now of growing interest when training spatiotemporal features from complex data channels [84, 85]. Here too, refining attention-based models might more effectively emphasize misclassified frames and segments via boosting, however, directly applying traditional boosting algorithms to deep neural networks is difficult due to computational complexity [86, 87].But success can still be achieved with these principles in hybrid approaches [88]. By enhancing challenging video sequences and focusing on eliminating temporal discrepancies, boosting enhances the robustness of ensemble models, especially given class imbalance and minor inter-class differences [89, 90]. Ensemble systems created by boosting are built sequentially whereby every new model learns to correct the previous models underlying mistakes [80]. Models included in the ensemble and being trained, use a weighted loss function that emphasizes, or gives greater weight to, harder-to-classify cases [18, 74, 75]. Boosting is useful for emphasizing rare patterns in videos, such as subtle actions or infrequent transitions [91].

At step t, the classifier f_t is trained to minimize a loss $\mathcal{L}$ on a weighted dataset:

$$\alpha_t = \arg\min_{\alpha} \sum_{i=1}^{N} w_i^{(t)} \cdot \mathcal{L}\left(y_i, f_t(x_i)\right) \tag{2.10}$$

where $w_i^{(t)}$ is the weight assigned to each training example. The final ensemble prediction is obtained via weighted combination of all learners as in Equation 2.11

$$\hat{y} = \arg\max_{y} \left(\sum_{t=1}^{T} \alpha_t \cdot f_t(x) \right) \tag{2.11}$$

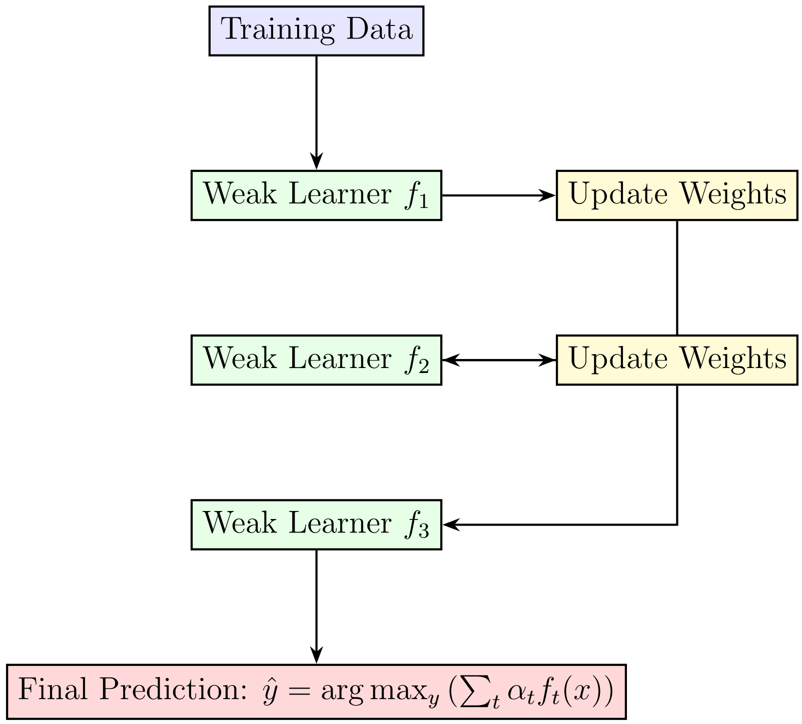


Figure 4: Boosting ensemble with sequential weak learners.

2.2.4.3 Stacking and Meta Learning

Stacking is an advanced ensemble method that seeks to build predictions based on the output of numerous models (called the base learners) by using a number of meta learners which learn the best way to combine base learner outputs [92]. Unlike bagging and boosting that rely on parallel or sequential training of models and simply voting or weighted majority aggregation, stacking follows a two-level learning paradigm (the base learners itself, and then the meta-learner) [18, 82]. The base learners produce predictions, which are then the inputs (features) to a meta-model that ultimately makes the final class decision [18].

The idea of stacking was coined and termed stacked generalization and its implementation improves the bias that exists by aggregating input from heterogeneous models [92]. Stacking these models in the context of deep learning is sometimes referred to as model blending, particularly when the final integrator is some simple linear model [92].The full architecture of the stacking ensemble is illustrated in Figure 5, where predictions from multiple base learners are fused and passed to a meta-learner for final decision-making.

Recent developments such as Deep Convex Networks (DCNs) implement the stacking idea whereby several distinct learning modules are structured and layered together, so that each module learns an abstract representation of the data before passing it to the next module down the stack [93, 94]. For video understanding, stacking can be beneficial when the different base models are focusing on different modalities, such as a spatial CNN, a temporal RNN, or an audio sequence [18, 92, 95].

Formally, if the base models produce prediction vectors $z_m = f_m(x)$, their outputs can be concatenated into a single vector $z = [z_1, z_2, \ldots, z_M]$, and passed to a meta-learner g, which produces the final prediction $\hat{y}$, as shown in Equation 2.12.

Let $f_1, f_2, \ldots, f_M$ be base models, each producing output vectors $z_m = f_m(x)$.

The meta-learner g is trained on the concatenated outputs:

$$z = [z_1, z_2, \ldots, z_M] \quad \Rightarrow \quad \hat{y} = g(z) \tag{2.12}$$

This enables the meta-learner to exploit relationships and complementarity between different models, modalities, or temporal resolutions.

Stacking is particularly useful in multimodal fusion tasks or in complicated and diverse video datasets where no single model is dominantly better than the others [95]. Stacking is a great ensemble method for dynamic video data, taking advantage of the combination of inductive biases present in its individual learners [18, 95].

Stacking trains together a number of base learners, and combines the outputs of the individual learners using a meta-learner. Stacking is different than bagging and boosting, as stacking can combine different model architectures and different modal features [18, 82].

Stacking is especially powerful for combining spatial, temporal, and audio features in video classification tasks.

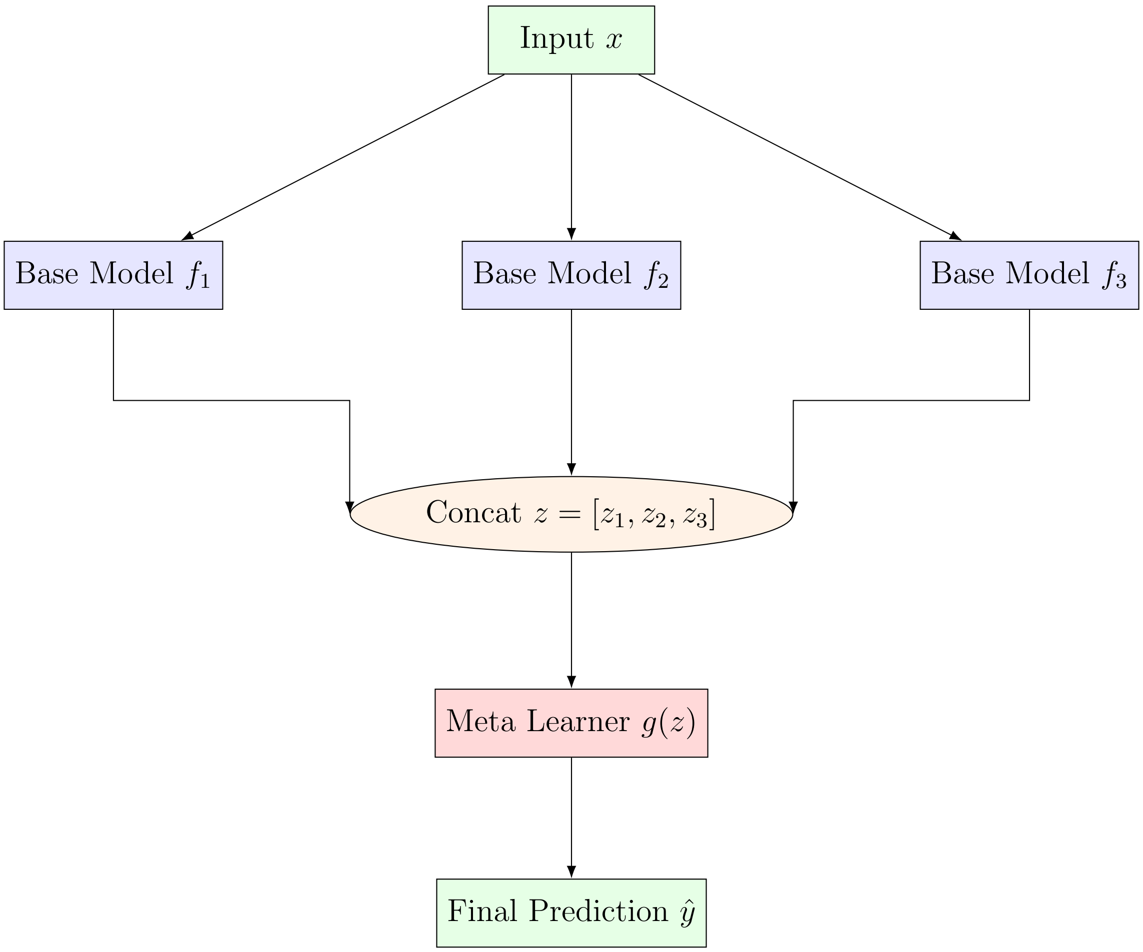


Figure 5: Stacking ensemble architecture.

2.2.5 Attention Mechanisms for Temporal and Spatial Modeling

These past few years, attention mechanisms are critical for aiding progress in video understanding tasks, with attention allowing models to focus or devote their processing capacity to only the most useful spatial or temporal parts of the full input sequences [96]. Attention mechanisms have key advantages in segmentation and classification tasks where intimately connected contextual relationships are essential on local or global scales [14, 96]. Convolutional architectures known to have difficulty computationally with maintaining long-range dependencies, while attention-based models leverage dynamic receptive fields based on feature importance [97]. In this section, we discuss important developments in attention mechanisms such as Region Attention networks, Axial Attention, and Criss-Cross Attention that strive to achieve a balance between inference costs and the need to model complex relationships in video data.

2.2.5.1 Region Attention Network

RANet (Region Attention Networks) proposed utilizing Region Attention as a way to add object-level context to semantic segmentation [98]. RANets consists of two modules only: the Region Construction Block (RCB) and the Region Interaction Block (RIB) [98, 99]. The RCB analyzes the boundary score maps along with the semantic score maps to compute region attention scores. The RCB groups together pixels likely belonging to the same object, based on their boundary properties [100]. The regions from the RCB are then passed into the RIB, where a small set of representative pixels within to the region are selected, allowing for indigenous pixel content but also allowing for communication between regions to help build a better understanding of global context [98, 101]. This hierarchical structure assists the RANet in maintaining spatial and semantic consistency across regions, which results a strong contextual feature map for use in segmentation [102].

2.2.5.2 Axial Attention

Axial Attention introduced by Wang et al. overcomes the computational inefficiencies of standard fully-connected self-attention by factoring 2D attention to two sequential 1D attentions along the height and width axes [103]. This factorization allows the model to efficiently capture long-range dependencies and global context while also decreasing the complexity from quadratic to linear [103]. Unlike previous approaches, axial attention uses explicit positional information, providing a more powerful way to model spatial structures. Axial attention achieves equal performance to models based entirely on self-attention but with significantly lower complexity. Axial attention offers a promising approach for cases where spatial accuracy is critical but may not be as restrictive as desired, such as in video segmentation.

2.2.5.3 Criss Cross Attention

Criss-Cross Attention is a step towards better contextual encoding for scene understanding tasks [104, 105]. Unlike non-local blocks, which attend to all spatial locations, Criss-Cross Attention attends to two paths of context for each pixel, one horizontal, one vertical. This leads to a substantial savings in memory—11×—and processing cost—85% GFLOPS—compared to original non-local modules [105]. The mechanism works by adding three 1×1 convolutions to the input features—two will be merged over the different rows and then softmaxed to produce attention scores and each row has corresponding attention values which will be merged with the third convolution output [105]. To avoid only encoding the criss-cross directional information, the authors proposed a recurrent Criss-Cross Attention module that implements the same attention mechanism repeatedly (with shared weights) to ultimately fully capture contextual information across the image [105]. By integrating both the efficiency of criss-cross attention and the global context of the image, this same design was able to be developed for dense prediction tasks such as scene segmentation in video.

The growth of attention mechanisms captures the overall shift towards context-indicated learning in video-based tasks [97]. Region Attention Networks build upon the concept of local object-level relationships through the construction and interaction of semantic regions; Axial Attention proposes a novel factorization of 2D attention into 1D axes as a means of scalable long-range contextualization; Criss-Cross Attention offers a memory-efficient option for global context to be aggregated through recurrent directional attention. When taken together, these methods demonstrate the ways in which researchers have explored diverse routes to challenge the limitations of early models in video, and suggest means to develop more efficient and semantically based video segmentation and classification systems. (Table 1)

2.2.6 Limitations of Current Approaches in Broadcast Video Contexts

Even though many advances have been made in video understanding, some of the recent methods for deep learning to model video content will have different weaknesses when we

Table 1: Overview of Attention Mechanisms for Semantic Segmentation

Feature	**RANet**	**Axial Attention**	**Criss-Cross Attention**
Key Idea	Object-level regions	Factorized 1D attentions	Recurrent directional attention
Efficiency	Region grouping	Linear complexity	Memory-efficient
Context	Local semantic regions	Long-range dependencies	Global context
Positional Info	Boundary-based	Explicit encoding	Implicit directional
Application	Video tasks	Video tasks	Video tasks
Original Paper	Shen et al., 2020 [98]	Wang et al., 2020 [106]	Huang et al., 2019 [104]

apply them to broadcast video content. Broadcast video is made up of specific content across different genres like news, talk shows, sports, live events that inherently challenges video understanding methods with things like: scene transitions that occur in a variety of ways; transient speakers interacting with each other; multiple camera angles serving different functions including visual transitions; and structured audio-visual cues both concluding and connecting sequences. Of course, for models based on CNNs, transformers — and general video tasks—the most success has come from tasks that have less variability in the structure and connection of content. Most state-of-the-art models are trained on short, trimmed clips that only last a few seconds which may result in models that cannot model long-term temporal relationships and are unable to maintain a coherent narrative across scenes. This is essential in the broadcast domain where videos may be hours long, involve plenty of cuts and switch topics regularly [107]. Often broadcast videos have sets of structure, for example, opening sequences, interviews, or panel discussions; state-of-the-art models do not have the ability to leverage those domain-specific structural patterns. Models trained on generic video datasets have no inductive biases to tell the difference between a news anchor's monologue or on-field sports commentary [108]. There are audio-visual models but many treat audio as an auxiliary signal (e.g. listen for audio in an example video) or use pretrained language models with fixed weights. Audio-visual models lack flexibility. In a broadcast context, speech cues, background sounds, and alignment of subtitles are critical for accurate classification of segments or understanding the semantic meaning of segments, but in the datasets I have reviewed, are used rather rarely [109]. Broadcast archives are often large and sparsely labeled. Many state-of-the-art models depend on dense annotations or heavily rely on supervised learning to be trained effectively. This creates the issue of requiring vast computational resources to scale to full-length broadcasts, while making the prospect of effective semi- or weakly-supervised learning unmanageable [110]. Video understanding systems did not work well in the complex, structured, and long-form nature of broadcast video. Their shortcomings in modeling long-range dependencies, processing rich multimodal cues, and transferring to domain-specific formats highlight the necessity for architecture and training paradigms specific to broadcast video [107, 108, 109, 110]. Solving this problem is critical not only to advance performance in broadcast settings but also to strengthen generalizable and real-world video understanding systems.

2.3 Related Work

Knowing that video classification has continued to evolve in the recent years, various approaches have been proposed to improve the model's efficiency, accuracy, and adaptability across diverse video content.

Video classification requires models to learn spatiotemporal representations from diverse sources of information. Knowing that researchers have explored a variety of techniques to improve both the accuracy and computational efficiency. In the following subsections, we review related works across four relevant areas: **scene segmentation and keyframe selection**, **attention mechanisms** in video understanding, **audio-visual and multimodal fusion techniques**, and **ensemble learning strategies** applied to video classification.

2.3.1 Attention Mechanisms

Attention mechanisms enable models to prioritize specific spatial and temporal features in video data. This focus allows better representation of relevant actions, objects, or scenes without relying heavily on sequential modeling.

Wang et al. proposed AttentionNAS, a neural architecture search (NAS) framework designed to discover spatiotemporal attention cells for video classification tasks [14]. Their approach addresses two major limitations of convolutional operations: the lack of explicit spatial focus and the inability to capture long-range dependencies. By designing a novel search space composed of attention-based operations, the framework enables efficient exploration of spatiotemporal attention cell structures. When integrated into backbone architectures like I3D and S3D, the discovered cells outperformed non-local blocks and achieved state-of-the-art performance on Kinetics-600 and MiT datasets, improving accuracy by over 2%. However, while the framework enhances global attention modeling, it does not explicitly focus on fine-grained object-level or region-specific attention.

Long et al. introduced Attention Clusters, a framework that abandons explicit temporal modeling and instead leverages purely attention-based local feature integration for video classification [13]. Motivated by the observation that local features in videos often exhibit redundancy, identifiability, and approximate temporal order invariance, the authors propose a method that aggregates informative frame-level features without relying on sequence modeling. Their approach introduces a shifting operation to enhance diversity in attention responses and demonstrates competitive performance across multiple datasets, including a top-1 accuracy of 79.4% on Kinetics. However, while the method effectively captures holistic patterns by summarizing local features, it lacks explicit mechanisms for structured attention—such as spatial focus on individual objects or action regions—which may limit its interpretability and fine-grained semantic understanding.

While both methods demonstrate how attention improves video classification by emphasizing informative frames or learning optimized attention structures, they focus primarily on overall video-level understanding. Their effectiveness in modeling object-specific or region-level attention remains underexplored—an important direction for tasks requiring finer spatial or temporal resolution.

2.3.2 Ensemble Learning Approaches

Ensemble learning has received much attention as an effective paradigm in video understanding aimed at improving model performance and generalization through the combination of predictions from multiple learners. Single-stream architectures in particular often have difficulty accounting for the inherent complexity and variability within video data, including factors such as spatial-temporal dependencies, dynamics between multiple objects, and varying ways to integrate data from multiple modalities. Ensemble learning could be effective in addressing all of these challenges simultaneously by leveraging model diversity from different learning models in a committee of learners combined with consensus building. Recently, ensemble methods have been utilized by more studies in tasks such as video classification, scene segmentation, and action recognition, with studies leveraging both implicit and explicit methods of combining additional information. This section describes representative studies employing ensemble-based strategies, with a focus on how the studies addressed challenges such as feature redundancy, modality imbalance, or trade-offs in computation time.

Chen et al. introduced GvSeg, a unified framework that can tackle multiple video segmentation tasks (instance, semantic, panoptic, and exemplar-guided segmentation) with a single design [111]. GvSeg not only addresses multiple segmentation tasks but also keeps them separate rather than treating all tasks the same. In other words, GvSeg disentangles the segmentation targets, encompassing appearance, position, and shape and modifies each segmentation strategy with respect to query initialization, matching, and sampling. This formulation of a framework allows the framework to adaptively develop a segmentation strategy without the need to change the overall architecture [111]. GvSeg reported state-of-the-art results over seven benchmark datasets, such as 4.8% AP on Occluded-VIS and 1.1% mIoU on VSPW. However, while the framework is undeniably general and demonstrates significant generalization across tasks, it may miss out on gains from potentially more powerful ensembles because it was built to form privately an ensemble of independently optimized models or representations for each segmentation task [111].

Xiangtai Li et al. introduced UniVS, a unified video segmentation framework that reimagines diverse segmentation tasks as prompt-guided target segmentation [112]. The architecture circumvents the challenge of handling both category-specified and prompt-specified tasks by using prompts as dynamic queries—averaging features from previous frames to initialize target-specific attention and leveraging a novel prompt cross-attention (ProCA) layer in the decoder [112].The UniVS treats previously predicted masks as visual prompts, effectively swapping our dependency on traditional inter-frame associations for a more adaptable, memory-oriented matching process. It shows excellent performance across 10 video segmentation benchmarks indicating both task generality and architectural simplicity. However, while UniVS simplifies segmentation paradigms through prompt conditioning, it does not have the complementary diversity that an ensemble-based design could offer - for example, as an augment to unify the outputs of different prompt types or purpose-built branches specialized for each segmentation sub-task [112].

Minghan Li et al. proposed OMG-Seg, a unified framework for segmentation that can handle over ten different tasks, including segmentation in images and videos (semantic segmentation, instance segmentation, panoptic segmentation, interactive segmentation and open-vocabulary segmentation) with just one encoder-decoder model [113]. It does so by using a unified query representation to encode task specific semantics masks, IDs, visual prompts, etc. so that the shared decoder can take different heterogeneous inputs with predictable regularity. This results in lower computational cost and cross-training across datasets becomes easier, facilitating stronger generalization. Nevertheless, OMG-Seg's reliance on one shared representation means it is making the compromise to balance flexibility and specialization [113]. Combining multiple expert models with ensemble approaches or combining adaptive submodules could allow more task specific inductive biases to be preserved, while maintaining cross-task synergy.

Ensemble learning remains vital in propelling the state-of-the-art in video understanding because of its ability to leverage diversity to result in robustness, which is important in reducing overfitting and encourages design and implementations of modularity. The varying strategies like feature-level fusion and multi-modal attention, along with task-unifying paradigms like UniVS and OMG-Seg, all show the potential of ensemble paradigms to unify distinct modalities or architectural components. Nevertheless, there are still several challenges—especially in gaining maximum efficiency with maximum accuracy, and composing ensembles that can scale across datasets and tasks. Future work can address opportunities like diversity maximizing, approaching self-supervised paradigms, and selecting among sub-ensembles dynamically or adaptively to capitalize on the momentum of large-scale video datasets.

2.3.3 Multimodal Fusion

Computer systems must grasp both visual sceneries and natural language inputs in order to engage with humans regarding visual information [8]. A system, in which the output of a visual description system is employed as an input, is one basic way [44] . Among the existing techniques, multimodal techniques [8] that use characteristics taken from many modalities produce the best results. Multimodal fusion can be achieved by fusing each modality as a single representation or by segmenting each modality, with the latter being more often used. Multimodal approaches make use of information derived from two or more modalities, ideally in a complimentary manner [8].

Aside from visual information, humans rely heavily on their ears to comprehend a certain scenario [114]. Audio is one of the most important aspects of video analysis and many studies have demonstrated that auditory and visual streams include complimentary information [115]. Audio features fall under either of time-domain and frequency domain [114]. Many low-level features have been studied, including short-time energy, zero crossing rate (ZCR), pitch, frequency centroid, spectral flax, and Mel Frequency Cepstral Coefficients (MFCC).

The latest revolution in neural network models, on the other hand, allows us to merge diverse modules into a single end-to-end differentiable network. Video information and human utterances can be input into an encoder-decoder system that generates natural language responses [44]. Deep learning techniques [8], like convolutional neural networks (CNN) have recently been used successfully to extract features from diverse data sources, facilitating the development of early fusion approaches.

Incorporating CNNs and RNNs into multimodal approaches has made both technologies very fascinating topics. RNNs in this domain were created first to process text sequence data. It is common usage to use LSTMs, a type of RNN, as well in the encoding and decoding networks of the latest models of audio-visual models for semantic segmentation [44, 50].

Hori et al. [44] explored scene-aware dialog systems that integrate visual features from pretrained VGG-16 and C3D models to generate spatiotemporal representations from video frames. They compared these with I3D features—derived from a two-stream inflated 3D ConvNet—on datasets like MSVD and MSR-VTT using natural language processing metrics such as BLEU and METEOR. Even though the I3D model was the best of the various combinations considered, it relied exclusively on visual modalities without incorporating audio signals, limiting its ability encapture the complete multimodal video context.

Carreira et al.[107], proposed an alternative version the I3D model, which employs an inflated 3D version of the Inception-V3 architecture. The I3D model was pretrained on the Kinetics dataset and obtained improved performance on action recognition than previous models, which were based on VGG-16. The model achieved improved accuracy at the same time as being computationally efficient and reduced the number of parameters, while at the same time, it wasn't able to utilize audio modalities, which limited its access to complementary multimodal cues in video classification and segmentation tasks.

Ye and et al. introduced a two-stream convolutional neural network (CNN) framework that separately represents the spatial information extracted from raw frames and the temporal dynamics obtained from stacked optical flow to effectively understand and classify video information [12]. In their study, the authors used a systematic process to assess the architectural differences, model fusion methods and prediction mechanisms, demonstrating that the a variety of ways to fuse spatial and temporal features can achieve state-of-the-art results of standard video datasets. Their work indicated that multimodal fusion is valuable in order to comprehend the complex semantics of a video clip, but the study is only using the development of a clip-level classification model as opposed to seeking finer level of understanding through video segmentation. The work outlined the usefulness of the two-stream CNN architecture, that combined spatial visual features and temporal visual features, for the purpose of video classification, but compared to other approaches, it did not incorporate audio or other non-visual modalities. This reduces the models ability to capture the full semantic context of the videos being classified in the event that audio cues existed that augmented, or even disambiguated, the visuals.

These limitations motivate the need for a more unified audio-visual framework that can leverage complementary modalities.

2.3.4 Video Scene Segmentation

One of these tasks is scene segmentation, which divides a TV series into its individual scenes. Berhe et al [32], worked focusing on developing an automatic system for segmenting scenes in TV series. Multimodal features extracted from deep neural networks are utilized to perform this task. According to the article [32], the experimentation used a multimodal feature of image and text. As of the case of Wu et al [113], even though multimodal features have been explored, the latest paper in the area, used video segmentation for the purpose of representing a scene consistency. The Self-Supervised Learning approach, which has been highlighted importantly, has been also included in the paper, which shows that the model can perform in scales outside the provided MovieNET datasets. The rift however is that both papers didn't use video scene segmentation for classifications tasks as is proposed by this paper.

2.3.5 Summary

Ensemble learning has made a considerable impact in video understanding projects, since it allows aggregating the predictions of multiple models to enhance robustness and generalization. Chen, et al. [111] proposed GvSeg, a unified architecture tackling a suite of different video segmentation tasks by disentangling the segmentation targets into components of appearance, position, and shape. Despite strong versatility and leading edge performance, GvSeg operates on an identical backbone, which may limit the models capacity to better fully leverage task specific optimizations that would normally be supported by explicit ensemble methods. Li, et al. [112] also proposed UniVS, and reliant on the notion that video segmentation can be reformulated as a prompt-guided task, motivated by prompts derived from previous frames, and implemented a new prompt cross-attention (ProCA) layer. Although it provides a general structure for segmentation as a whole, the UniVS lacks representational diversity by not including any variety if prompt types or pooling from specific branches for each subtask. OMG-Seg, which was presented in Minghan Li et al. [113], is a unified segmentation network that supports over ten unique segmentation tasks within a single encoder-decoder model using a unified query representation. The unified representation allows for efficient co-training and strong generalization—however, it sacrifices flexibility and specialization. Ensemble-based approaches, which instead could better exploit inductive biases unique to each task while still benefiting from cross-task cooperation.

At the same time, multimodal fusion approaches are emerging an important direction in video understanding, as they seek to fuse complementary pieces of information in visual and auditory cateogories. In as mentioned in [8], systems can be improved by utilizing additional modalities in complement, and therefore these fusion approaches can significantly improve performance. Hori et al. [44] developed an end-to-end audio-visual scene-aware dialog system that processes video information alongside human utterances using an encoder-decoder architecture. While their approach integrates spatiotemporal visual features from pretrained VGG-16, C3D, and I3D models, it omits audio signals, limiting its capacity to capture a fuller context of multimodal video data. In the same vein, An inflated 3D ConvNet (I3D) was proposed by Carreira et al. It was based on Inception-V3 and was pretrained on Kinetics [107]. While the model made considerable performance gains in action recognition, it also does not use audio cues and its application to certain contexts that require coordinating the audio-visual channel was limited. Ye et al. introduced a two-stream CNN that utilizes raw frames and optical flow to treat spatial and temporal information separately [12]. The proposed methodology was competitive for video classification, but actively limited the scope of study to visual inputs and clip-level classification while leaving multimodal fusion and more fine-grained scene-level tasks unexplored.

Building on this, scene segmentation has emerged as a foundational task for structuring long-form video content into semantically coherent scenes. Berhe et al. [32] proposed an automatic system that segments TV series using multimodal features—specifically image and text—extracted via deep learning. Although useful in boundary detection, this method does not go beyond this to downstream tasks such as scene-level classification. Productions by Wu et al. [113] included desired temporal consistency by leveraging video segmentation across scenes and used self-supervised learning for generalization beyond the MovieNet dataset. However, neither work mentions to merging scene segmentation and classification tasks, pointing towards a gap in utilizing segmented outputs for even higher-level semantic inference, pointing to the opportunity for frameworks that can segment and classify jointly, perhaps in an ensemble manner using multimodal inputs, for semi-brittle and scalable video understanding.

Table 2: Summary of Related Work and Identified Gaps

Author (Ref.)	Category	Reported Metric	Identified Gap (Generalizability, Modality, Efficiency, Label Dependence)
Chen et al. [111]	Ensemble Video Segmentation	AP: +4.8%	Highly versatile, but unified backbone limits specialization. No multimodal inputs are used, and high supervision limits scalability.
Li et al. [112]	Prompt-Guided Segmentation	Qualitative	Prompt-based design generalizes segmentation tasks, but lacks diverse prompts and modality fusion. Moderate efficiency, some label dependence.
Li et al. [113]	Unified Video Segmentation	Efficiency Focused	Enables efficient co-training but lacks flexibility and specialization. Modality fusion is not present.
Hori et al. [44]	Audio-Visual Dialog / Fusion	BLEU: 0.45	Integrates visual and text signals, but omits audio and segmentation tasks. Generalization beyond dialog is unclear.
Carreira et al. [107]	Action Recognition	Top-1 Acc: 79.8%	Effective for action recognition with visual cues only; lacks multimodal input and scene understanding capabilities.

Author (Ref.)	**Category**	**Reported Metric**	**Identified Gap (Generalizability, Modality, Efficiency, Label Dependence)**
Ye et al. [12]	Two-Stream CNN Classification	Acc: 85.7%	Combines visual spatial and temporal streams. However, it excludes multimodal fusion and is limited to clip-level tasks.
Berhe et al. [32]	Multimodal Scene Segmentation	Qualitative	Uses image and text features for boundary detection but does not support scene classification or generalized downstream tasks.
Wu et al. (in Li et al.) [113]	Scene Segmentation + SSL	N/A	Applies self-supervision and achieves generalization, but lacks multimodal fusion and downstream classification capabilities.
Multimodal Overview [8]	Fusion Techniques Survey	Lit. Review	Conceptual insights into modality fusion, but no empirical methods. No contribution to computational efficiency or label reduction.
You et al. [114]	Audio Feature Extraction	Conceptual	Focuses on audio alone. No visual context or segmentation utility. Outdated for modern multimodal learning.

Author (Ref.)	**Category**	**Reported Metric**	**Identified Gap (Generalizability, Modality, Efficiency, Label Dependence)**
General Survey [115]	Audio-Visual Complementarity	Lit. Review	Highlights theoretical complementarity, but lacks practical implementation and evaluation for segmentation or efficiency.

Note: Gaps are defined across dimensions of generalizability, multimodal integration, efficiency, and annotation dependence.

Chapter 3 Methodology

3.1 Overview

The overall purpose of this research is to create an accurate video classification system with an attention based visual model as well as an auditory classifier. The proposed method that we are going to design is a dual-stream model combining video and audio streams for additive classification performance through the benefits of both visual and audio modalities. The system shall consist of two source components:

1. **Visual Stream**: This will take the video frames, and through attention and object detection, generate high-level spatial and semantic features. Attention will allow the model to attend the most salient parts of the image, while object detcetion will account for which entities are relevant to the scene [116, 117].

2. **Audio Stream**: The audio sub-system will operate a ConvLSTM-based classifier as the system will model temporal dynamics and capture relevant acoustic features. This system model lends itself to working with non-stationary audi signal with robustness in sequential classification tasks involving audio [118, 119].

The outputs from both streams are subsequently fused to enable joint decision-making, with the expectation that combining visual and audio cues will result in improved classification accuracy compared to unimodal approaches. The first stage of the system is dedicated to the analysis and interpretation of visual information from video frames. This stream, referred to as the *visual stream*, is designed to extract program-specific visual features typically found in the initial seconds after a broadcast begins—such as program logos, channel identifiers, or branded intro sequences. It is composed of two integrated components. A deep-learning object detection model, YOLOv3 [117], enhanced with attention mechanisms, is used to detect pre-defined program-centric visual cues. The model is trained on a carefully constructed dataset of visual markers that includes logos, opening frames, and other brand identifiers. The attention layers help the detector to focus on the most semantically relevant parts of the frame, and improve precision and robustness to background noise [116, 120]. The system has a fallback architecture and a joint ensemble approach. In fallback mode, the visual stream is the primary decision-making stream, while the audio classifier is activated only in instances in which the visual stream has uncertain or noncommittal predictions [121, 122]. Fallback mode means performance is prioritized at the expense of interpretability. Then, in ensemble-based mode, visual and auditory nodes will be attended to simultaneously. Eventually, the outputs of both the visual and auditory nodes in the ensemble-based mode will get integrated at either the feature or decision level for classification [123, 124], as shown in the Algorithm 2. This two-stream ensemble mode uses the virtues of each modality in concert, and can be expected to increase robustness with respect to occlusion, low auditory clarity, or uncertainty in visuals [125]. Experiments are being conducted in both modes across various broadcast scenarios, aiming to determine the optimal integration strategies based on performance, resource constraints, and failure modes. A visual summary of the proposed processing pipeline is shown in Figure 6, where the sequential object detection and multimodal fusion process is outlined. The segmentation and classification logic, particularly the attention-guided modeling of video structure, is formalized in Algorithm 1.

Algorithm 1 Attention-Augmented Video Object Detection

Require: Video frames $V = \{F_1, F_2, ..., F_T\}$, Backbone CNN B, Positional Encoding function PE, Attention module $\mathcal{A}$, Object Detector $\mathcal{D}$

Ensure: Detections $D = \{D_1, D_2, ..., D_T\}$ where D_t contains bounding boxes and class scores for frame F_t

for each frame F_t in V **do**
    Extract visual features: $f_t \leftarrow B(F_t)$
    Add temporal/spatial encoding: $e_t \leftarrow f_t + PE(t)$
end for
for each encoded frame e_t in $\{e_1, ..., e_T\}$ **do**
    Compute attended features: $\hat{f}_t \leftarrow \mathcal{A}(e_t, \{e_i\}_{i=1}^{T})$
    Detect objects: $D_t \leftarrow \mathcal{D}(\hat{f}_t)$
end for
return D

The system is built upon a multimodal ensemble framework, which integrates visual and audio in order to strengthen the robustness of the classification. Initial base classifiers will be utilized to classify their modality independently in the first stage. Their predictions will then be conjoined and accepted by a meta-classifier which will provide the final prediction as shown in Algorithm 2.

Algorithm 2 Multimodal Ensemble Learning with Audio

Require: Visual features V, Audio features A, Video classifiers $\{C_v^i\}_{i=1}^n$, Audio classifiers $\{C_a^j\}_{j=1}^m$, Fusion function $\mathcal{F}$, Meta-classifier G

Ensure: Final prediction $\hat{y}$

for each visual classifier C_v^i **do**
    Compute visual prediction logits: $p_v^i \leftarrow C_v^i(V)$
end for
for each audio classifier C_a^j **do**
    Compute audio prediction logits: $p_a^j \leftarrow C_a^j(A)$
end for
Fuse all logits: $z \leftarrow \mathcal{F}(\{p_v^i\}, \{p_a^j\})$
Predict final output: $\hat{y} \leftarrow G(z)$
return $\hat{y}$

The ensemble model detection is trained on a dataset of video clips that contain intro, credit and outro sections. The models can identify the intro and outro sections in a video clip and return the start and end times of the sections. The ensemble of models is able to segment video scenes with high accuracy. The methodology is simple to implement and can be easily adapted to different types of broadcast programs and even replicated to commercials as well. The methodology section of this paper will describe the steps taken to develop a model that can segment scenes in videos.

3.2 Audio-Based Classification using Convolutional LSTM

The audio stream processes audio segments with synchronized audio processing alongside the visual stream. Audio clips in their raw state are converted to log-Mel spectrograms, which provide input representations which maintain frequency and time aspects which are relevant for classification purposes.A Convolutional Long Short-Term Memory (ConvLSTM) network then operates on these features, probing the audio signal for short-term spectral variations coupled with longer temporal dependencies.

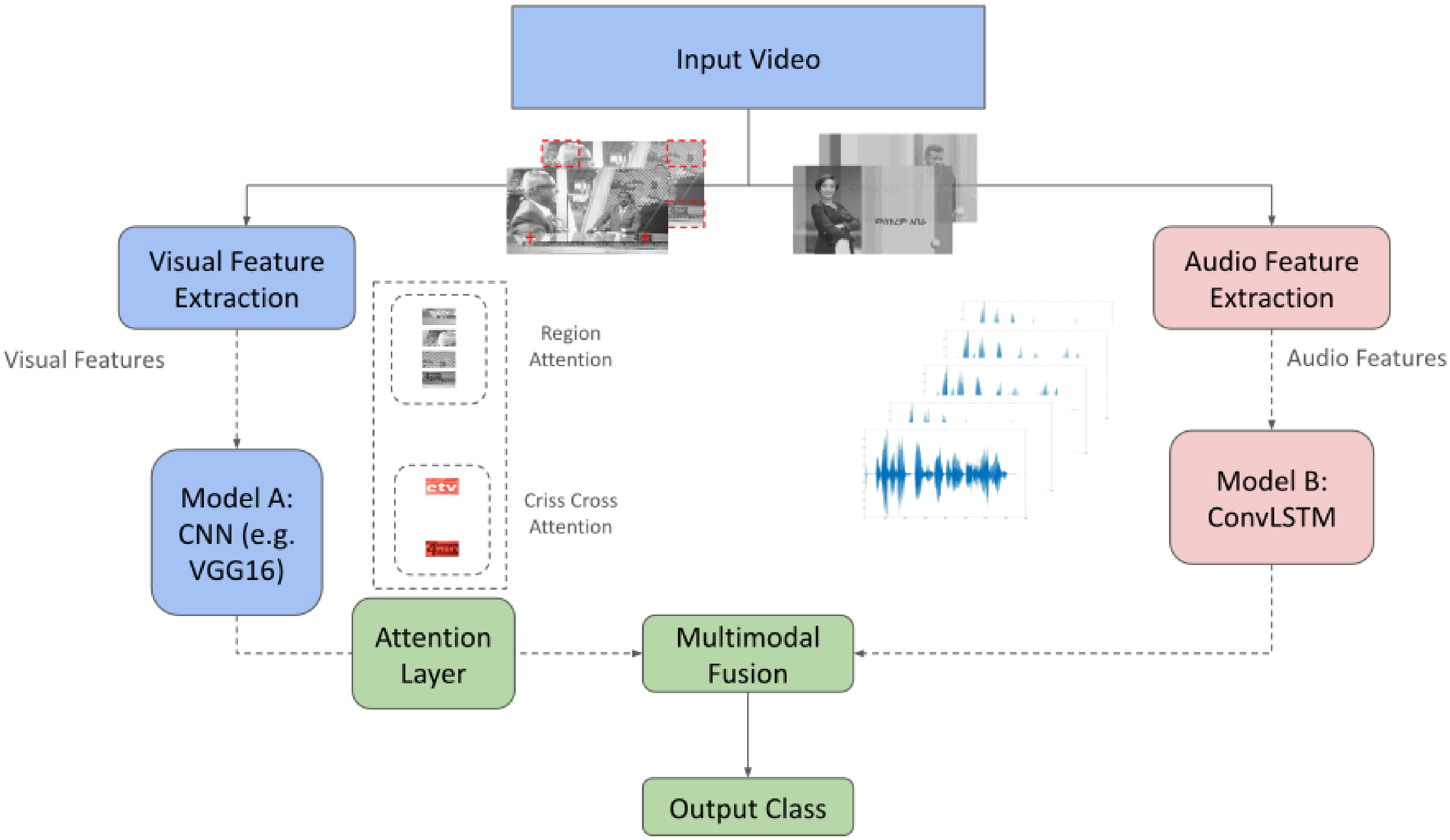


Figure 6: Methodology Overview

The ConvLSTM system is trained to classify video scene segments based on certain distinctive acoustic patterns both unique to each segment, like the patterns of the opening and ending musical sequences, patterns during segments of dialogue, and the music played throughout the rest of the program.This modality is very useful in contributing towards a more robust classification, especially when visual signals get difficult to see, get occluded, and/or are unreliable. In analyzing concurrent modalities, it is important to understand that temporal modelling is combined with convolutional feature extraction in order for the audio stream to assist the video stream, which, in our case, is also served as a part of the ensemble for decision-making [118].

To fuse these two streams, an ensemble strategy is employed.

The two streams—visual and audio—are integrated with a soft-voting ensemble. In doing so, each model is contributing a confidence score to the final prediction, and these scores are included with different weights, indicating which of the streams is most reliable. Ultimately, the final prediction is a probabilistic combination of visual and audio signals.
However, should the predictions from the two streams differ markedly, the system favours whichever model had stronger temporal consistency amongst adjacent frames—this provides temporal stability to the predictions and circumvents the compromises made with the model being subject to noisy input or scene ambiguity. The combination of all the models allows the system to make highly accurate predictions across multiple content types to systems that are trying to adapt to a wide range of lighting conditions, occlusions, and broadcast styles.

3.3 Selective Activation

The proposed method also addresses the computational efficiency issue by using a form of selective activation. Instead of passing each sequential frame through the complete network processing pipeline, it proposes the use of keyframe extraction techniques to only extract the most representative frames of the clip. This mitigates the effect of redundant information and reduces processing overhead.
After keyframe sampling the pipeline uses conditional routing to determine which components to enable. The first few frames will apply the object detection model to examine if there were identifiable logos or visual markers. If a high-confidence detection is achieved, the system proceeds with minimal computation. However, if the detection confidence falls below a predefined threshold, the more resource-intensive attention-based visual model and the audio classifier are subsequently activated. This approach ensures that the system uses the lightest possible inference path while preserving classification performance and multi-modal understanding.

3.4 Dataset Collection

Data for this research came from a range of sources, including local media companies in Addis Ababa's compliance records, publicly available videos on YouTube, as well as other streaming video content. The compliance records were good quality videos, and provided quality video and audio data with reasonable reliability for resolution and output performance. In contrast, the YouTube videos and other streamed content were used to supplement the compliance records and contribute a broader range of data. Data collection involved several approaches. The research team contacted local media companies to request access to their compliance records. In addition, they searched for publicly available videos on YouTube and other sites to stream the videos; these videos were then downloaded and stored on a local machine. Once the videos were downloaded, the data was cleaned and prepared for further use.This involved converting the videos to a common format, removing or anonymizing any identifying information, and segmenting the videos into smaller, meaningful scenes for analysis. The following table summarizes the dataset collection process. It includes the channel, the length of the individual video, the image numbers that are annotated and the source that it has been found, as shown in Table 3.

Table 3: Dataset Summary by Channel and Program

TV Channel	Program Name	Total Duration (HH:MM:SS)	Source Type
EBC	EBC Maezen	00:04:52	Public Archive
	EBC News	00:38:49	Public Archive
	EBC57	00:03:40	Public Archive
	EBC Sport	00:13:10	Public Archive
	Keyy Mesmer	00:46:12	Public Archive
FANA	Fana90	00:05:00	Compliance Record
	Fana Sport	00:04:59	Compliance Record
	Zetenegnaw Shi	06:46:49	Compliance Record
	Fana Kelemat	00:05:00	Compliance Record
Nahoo	Medlot	01:12:52	Compliance Record
EBS	MinLitazez	02:55:25	Public Archive
	BeSintu	05:32:56	Public Archive
	YeBeteseb Chewata	00:44:30	Public Archive
Total		**18:24:14**	

The first stage of the data collection process was dedicated to training the object detection model. During the annotation process, a straightforward visual analysis was conducted to identify specific elements within the video frames. This involved recognizing distinguishing features such as the logo of the television channel and the emblem associated with particular programs. These two visual elements—the TV channel logo and the program emblem—were the primary objects annotated for use in training the model.

Table 4: Summary for Objects on Dataset

TV Channel	Program News	Channel Logo	Program Emblem	Special Details
EBC	EBC Maezen	✓	✓	
	EBC News	✓	✓	
	EBC57	✓	✓	
	EBC Sport	✓	✓	
	Keyy Mesmer	✓	✓	
FANA	Fana90	✓	✓	
	Fana Sport	✓	✓	
	Zetenegnaw Shi	✓	✓	
	Fana Kelemat	✓	✓	✓
Nahoo	Medlot	✓		✓
EBS	YeBeteseb Chewata	✓		✓

3.5 Pre-Processing and Content Clarification

After the data set had been compiled, additional preprocessing, initial analyses, and meaning verification were required prior to training models. The initial step was a complete cleanup of the data, which included inspection and deletion of errors and inconsistencies. The images had been annotated with a variety of tools, and compiled using a single file and format to address compatibility in the model training and evaluation processes. The next step addressed verification of the meaning of the data as well as identification of unclear and/or ambiguities. The following section summarizes an experiment that involved training a sequence model to classify certain aspects of the data within a subset of the dataset containing credit, intro, and outro segments. During this initial stage of experimentation, all testing and model training were conducted solely for static image classification.

3.6 Scene Representation

The scene representation is a critical part of the idea of the paper. The approach of the representation on the mentioned experimentation result is very important. A single scene of the intro of the MinLitazez for example can be represented by consecutive images and a small audible catchy song that probably won't last for more than 15-20 seconds.

The idea is to represent a single scene, with consecutive images and a running audio without annotating any specific object that or emblem like we did previously. Preliminary tasks have been performed in order to standardize and represent a certain scene with both audio and image features. The images of a certain 20-30 second scene vary depending on the scene change, but images from the video has been captured within a single frequency per second (1fps) and the audio is represented by an mfcc descriptor which has 13 descriptors and a window size of 512. After that the mfcc is further standardized by taking a clustered points of 500 centers. The image on the other hand, a simple convolutional feature extractor using a VGG model was used.

The table below shows a sample of training data that was used for experimentation.

Table 5: Sample Segmented Scenes Used for Audio-Visual Experimentation

Name	**Section**	**Scene Number**	**Audio Length (s)**	**Total Images**
MinLitazez	Credit	3	00:48	49
ZetenegnawShi	Intro	4	00:16	18
BeSintu	Outro	7	00:16	16

As illustrated in Figure 6 the object detection model, illustrated by the above experimentation is limited to the defined data. If there is no emblem or logo or a special scenery that should be captured the models' output will soar down. The need for another model, regardless of a specific object is needed. The next experiment further clarifies how the merging of a CNN and RNN model is also essential in capturing and classifying an image and also audio altogether. The problem is designed to further accentuate the dynamics of the respective model and also the model that has been experimented above. Accordingly, the dataset has been further explored on those video sets that don't have emblem or logo.
The table below shows what has been used for the second experimentation.
The experimentation further explored how to represent specific parts or scenes within the dataset. As previously mentioned, the second phase of the experimentation focused on the Intro, Credit, and Outro segments of a television show, framing these elements within the context of a video segmentation task. The Intro refers to the introductory section that appears before the show begins, typically featuring producers and filmmakers, often accompanied by logos and names. The Credit segment follows, which—particularly in the case of sitcoms—marks the start of the program and includes the introduction of celebrities or key characters. Finally, the Outro denotes the closing portion of the video, signifying the end of the show, where credits are displayed and the screen fades to black.

3.7 Annotation Challenges and Motivation for Multi-Modal Representation

Throughout the initial experiments, it became apparent that the only use of annotated visual elements - i.e. emblems, logos or unique scenery - was insufficient to facilitate consistent scene identification for a particular scene. It became evident throughout these experiments that scenes often did not have an identifiable object that could be selected, reducing the potential success of the object detection model. The practical problem was that the understanding of the process was too reliant on adequately annotated object data.

To address this, the experiments proceeded in representing scenes not by specific annotations of objects but through a series of consecutive images and their audio data. This enabled the scene could be classified on more general attributes, even if less specific detail was derived from the visual component. The model was not focused on describing what the object is, but learning to indicate the rhythm, transitions, and structure of a scene.

There was not only a reduction in manual annotation effort, but also better representation of scenes that depended on broadcast type patterns, through introductions, credits or outros, where the audio-visual flow contained more contextual information than the static visual itself.

Chapter 4 Experimentation

This chapter presents the details of the experimental setup, including the tools used to develop and evaluate the models, as well as the performance outcomes of the proposed system. The final part of the chapter provides a discussion on the system's effectiveness based on the results obtained.

4.1 Tool Usage

The prerequisite programming environment for this study has been reviewed. The programming environment was tailored to set up the study and to implement the solutions suggest by the proposed models. Python version 3.10 was selected as the programming language due to its flexibility, library support, and its applicable to the machine learning and multimedia processing domain.

Various libraries and tools were taken advantage of throughout the experimentation phase. The NumPy library was being utilized for scientific computing (numerical data) during experimentation. Librosa, is an audio analysis library that provided many features for analysis and also feature extraction and processing of digitally represented signals. The TensorFlow library was effectively the basis on which the machine learning models were built and trained, whilst also managing the processes involved in the computational graph that supports deep learning. The SpeechRecognition library alongside variations of methods developed the speech to text segments of the audio processing components of the system.

The serialized data was handled using the Pickle library (which saves and loads Python objects). Moreover, for working with structured data and data analysis, we chose to use Pandas since it provides powerful data structures and operations. The image data was visually annotated using labelImg, which is a graphical image annotation tool that allows the manual labeling of visual objects for object detection work. Finally, we used FFmpeg, a popular, free open-source multimedia framework, to convert and process multimedia files. FFmpeg allows us to ensure that video and audio files are compatible and uniform.

4.1.1 Feature Extraction

This stage of the experiment focused on extracting relevant features from both audio and image data to enable multimodal learning through the ensemble model.

4.1.1.1 Audio Feature Extraction

The audio preprocessing pipeline employed the Mel Frequency Cepstral Coefficients (MFCC) algorithm, a widely adopted method for capturing perceptually relevant spectral characteristics of audio signals. MFCCs were chosen for their proven effectiveness in modeling the human auditory system by compressing frequency information according to the mel scale, which mirrors human sensitivity to sound [126]. This makes MFCCs especially suited for speech-related tasks and environmental sound classification in noisy conditions, as found in broadcast media. Compared to alternatives like log-mel spectrograms or raw waveforms, MFCCs provide a more compact and interpretable representation while maintaining strong discriminative power [127]. After generating the MFCCs, dimensionality reduction and normalization were performed using a K-Means clustering model, which reduced the feature set to a consistent structure comprising 200 representative temporal frames. Each frame consisted of 13 coefficients, resulting in a final feature vector of shape $(200, 13)$ for every audio segment.

4.1.1.2 Image Feature Extraction

The visual feature extraction process began with resizing each image frame to a resolution of 28×28 pixels. These resized frames were converted into NumPy arrays and subjected to normalization based on their RGB channel statistics. The normalization process was conducted channel-wise, using the mean and standard deviation of pixel values for each color channel. The normalized pixel intensity $NI(x, y, c)$ at spatial coordinate (x, y) and color channel c was computed as:

$$NI(x, y, c) = \frac{I(x, y, c) - \mu_c}{\sigma_c} \tag{4.1}$$

where $I(x, y, c)$ is the original pixel value, μ_c is the mean RGB value for channel c, and σ_c is the corresponding standard deviation. This normalization ensures consistent intensity distributions across the dataset.

Following normalization, pixel values were scaled to the range $[0, 1]$ by dividing each element of the image array by 255. The processed image arrays were then passed through the VGG16 convolutional neural network to extract deep visual features. The final output for each frame preserved the three-channel spatial structure with a feature vector dimension of $(28, 28, 3)$, suitable for subsequent fusion with the audio features in the ensemble model.

4.2 Ablation Study

4.2.1 Experiment 1: Visual Stream Analysis

This experiment aims to evaluate the impact of different training configurations on object detection performance, forming a visual baseline for the multimodal system. This ablation study analyzes how different training configurations impact object detection performance. The study was divided into three distinct components consisting of the evaluation of different model configurations, the use of specific tools/frameworks used in the development, and the in-depth analysis of the results. The evaluation was conducted on two different datasets and different training parameters, to look at the accuracy and robustness of the object detection model.

4.2.2 Evaluation

After processing, the annotated image dataset was employed to train a YOLO-based object detection model. The model was trained while utilizing an SGD optimizer and a batch size of 16, an input resolution of 640 pixels, a learning rate of 0.01, an IoU threshold of 0.7, and a weight decay of 0.0005. In order to track the model's learning behavior over its lifetime, we set up two experiments depending on the length of training. In the first experimental setup, the model was trained for 3 epochs, which took approximately 30 minutes on a computer with an NVIDIA GeForce RTX 3050 Ti Laptop GPU (4096 MiB RAM). In the second

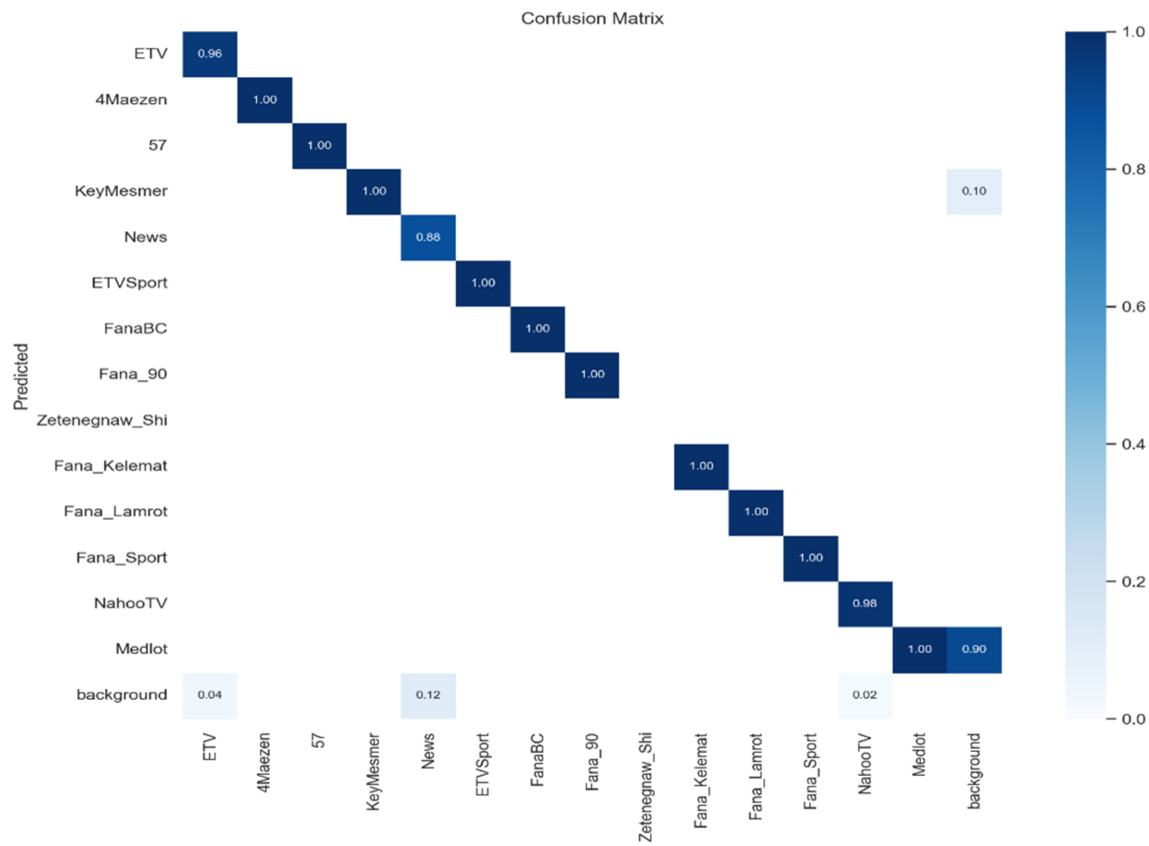


Figure 7: Confusion Matrix

experimental setup, we increased the epochs to 100, which took approximately 8 hours on the same hardware.

4.2.3 Results

The outcomes of the experiments, conducted with the previously described hyperparameters, demonstrate how the model's performance evolves with extended training. In the first setup with only 3 epochs, the training loss for bounding box prediction was reduced to 0.915, and the precision score reached 0.1679, based on the mAP@50 metric. These initial results reflect the model's basic capability to identify and localize objects with minimal training time.

The bounding box train loss represents the model's ability to predict accurate object locations; a lower loss indicates greater localization accuracy. Similarly, the precision score evaluates the accuracy of positive predictions, and in this case, the modest value is acceptable given the limited training duration. Training for a shorter period can be advantageous if the model is able to capture relevant patterns in the data quickly.

As summarized in Table 6, the model trained for 100 epochs yielded substantially improved performance. The training loss dropped to 0.1385, and the precision score increased to 0.18912. Both values indicate enhanced object localization and classification accuracy. Additionally, the F1-Confidence curve for this extended training setup exhibited superior performance across nearly all labels, demonstrating improved generalization and confidence in the model's predictions.

Table 6: Comparison of Object Detection Performance Across Epochs

Epochs	**Training Loss**	**Precision (mAP@50)**	**Training Time**
3	0.915	0.1679	˜30 minutes
100	0.1385	0.1891	˜8 hours

The F1-Confidence curve illustrates the balance between precision and recall at various confidence thresholds. Higher F1 and confidence scores observed in the 100-epoch model suggest both improved accuracy and increased certainty in predictions. This improvement affirms that longer training durations allow the model to better capture the underlying data distribution, resulting in more reliable object detection.

The comparison between both training configurations was further evaluated using precision-confidence, recall-confidence, and precision-recall curves. These metrics collectively offer a comprehensive view of the model's behavior under different classification thresholds. Precision-confidence curves reflect the model's accuracy at varying confidence levels, while recall-confidence curves measure completeness of predictions. Precision-recall curves depict how the model balances false positives and false negatives, revealing sensitivity to threshold changes as illustrated in Figure 11.

The graphs confirm that the model trained over 100 epochs outperformed the one trained for only 3 epochs across all key metrics. The later configuration not only achieved higher precision but also demonstrated improved recall and more consistent confidence levels, reinforcing the benefits of prolonged training.

Qualitative evaluation was also conducted through visual inspection of the model's predictions. As shown in the subsequent figures, the model effectively detected logos and other relevant objects from various TV program batches. The precision improvements are visually evident, especially in the higher-epoch training results.

Further visualizations illustrate model performance across different television channels and program types. These batch-level predictions reinforce the model's ability to generalize across diverse content, a critical capability in multimedia analysis tasks.

In summary, this ablation study demonstrates that extended training duration significantly enhances model performance across both quantitative metrics and qualitative evaluations. The improvements in bounding box accuracy, precision, F1 score, and visual consistency all support the conclusion that training with more epochs enables more robust and reliable object detection in multimedia data.

4.3 Effect of Attention Mechanisms on Object Detection Performance

To assess the contribution of attention mechanisms in the object detection pipeline, we introduced two attention-based modules—Region Attention and Co-Axial Attention—on top of the pretrained model trained for 100 epochs using the following hyperparameters: Adam optimizer, learning rate of 0.001, batch size of 64, Sparse Categorical Cross Entropy loss, and Accuracy as the evaluation metric.

These modules were used post-training, inference only, not fine-tuned or backpropagated after the attention was applied, putting us in a position to assess if attention can improve interpretability and localization, in an efficient and non-intrusive manner.

Region Attention improves the feature map by intensifying semantically meaningful spatial locations, while Co-Axial Attention merges local channel-wise attention while applying

Table 7: Comparison of Object Detection Performance With and Without Attention Modules

Model Variant	**Precision (mAP@50)**	**Loss (Train)**
Baseline (No Attention)	0.18912	0.1385
Region Attention (Inference)	0.19274	0.1385
Co-Axial Attention (Inference)	**0.19801**	0.1385

global, spaced the attention to both axes, enabling the capture of long range dependencies without the computational burden.

4.3.0.1 Quantitative Evaluation

The results are summarized in Table 7. The original model (without attention) achieved a precision of 0.18912 with a training loss of 0.1385. Introducing Region Attention slightly improved precision to 0.19274, while Co-Axial Attention provided a more substantial boost, reaching 0.19801. The loss remained identical, indicating that attention mechanisms primarily refine interpretive precision rather than altering optimization behavior post hoc.

Figures 12 and 13 provide qualitative and curve-based insights. Visually, the Co-Axial Attention model produced tighter, more accurate bounding boxes, especially in scenes with multiple overlapping objects or low contrast. Region Attention also provided cleaner separations, although slightly less precise than its coaxial counterpart.

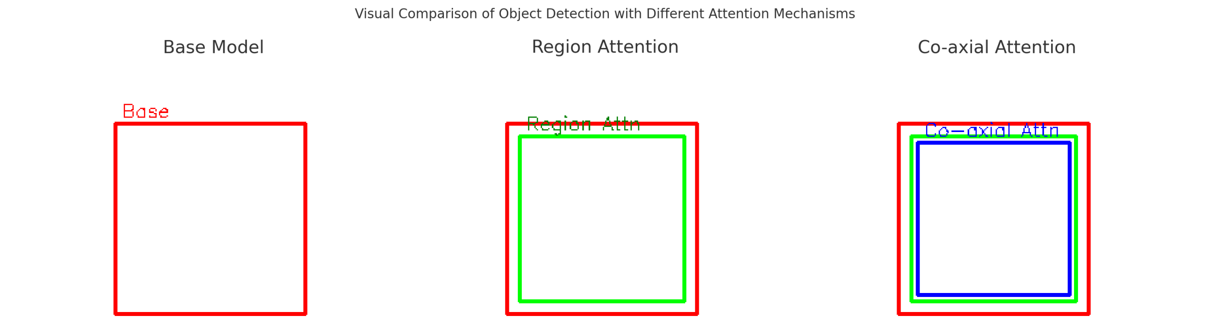


Figure 12: Bounding box results: (a) No Attention, (b) Region Attention, (c) Co-Axial. Co-Axial improves localization.

4.3.0.2 Curve-Based Evaluation

Precision-recall and F1-confidence curves were plotted to analyze performance over a range of confidence thresholds. The Co-Axial Attention curve maintains higher values across thresholds, while Region Attention offers moderate gains over the baseline.

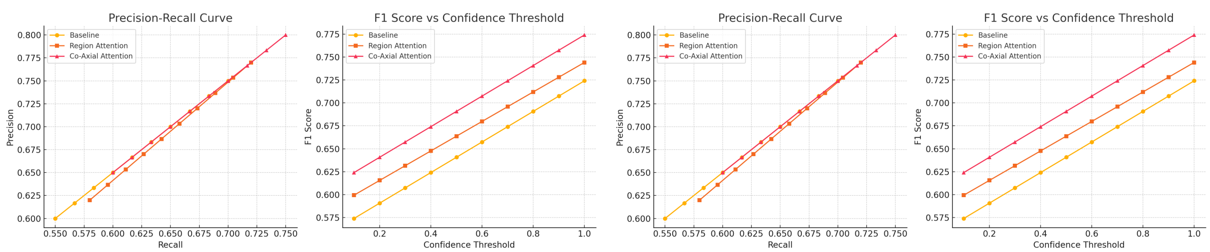


Figure 13: (Left) Precision-Recall curves, (Right) F1 vs Confidence. Co-Axial consistently outperforms others across thresholds.

4.3.0.3 Interpretation

These findings validate that attention layers can assist the object detection model in becoming more certain, and accurate, without performing further training. This demonstrates a low-cost, post-hoc enhancement for deployed models in the real-world, where it is unrealistic to do further training. Co-Axial Attention has good complexity/accuracy tradeoffs, and leads to improved resilience of cluttered and varied visual input.

4.3.0.4 Relevance to Multimodal Architecture

This visual stream analysis forms a critical foundation for the broader multimodal classification system proposed in this study. By first validating standalone visual performance, we establish the groundwork for audio integration in subsequent experiments.

4.4 Experiment 2: Ensemble Model

4.4.1 Evaluation

The second experiment was designed to test an ensemble model using segmented audiovisual data. Preprocessing was done from scratch and involved the extraction of both audio and image data from the raw video. As described in the methodology section, this experiment focused on the intro, credit, and outro segments. These segments were targeted because of their reliable structural markers and utility for recognizing and segmenting programs.

After extracting the scenes with the relevant performances in play, it was again necessary to refine the extracted frames to remove black screens in transition or associated with unused audio, silent audio, or irrelevant audio. Improving the sample quality was important to increase their usability for training and prediction. The original video was processed at a rate of 1 frame per second, essentially giving 1:1 sample rates on image and audio segments. The still frames were saved as PNG files to keep the quality of the images from degrading. The audio files were then converted into a 16-bit WAV substitute still using a PCM signed 16-bit little-endian codec. The audio had a sampling frequency of 16,000 Hz, with the convention of two channels (stereo) used. Further, the length of the videos and audio segments were all standardized to one second in the study to provide consistency across the dataset and allow the multi-modal system to learn in a synchronized manner in the ensemble method.

4.4.2 Ensemble Model Types

In this experiment, the ensemble model is implemented through **feature-level (early) fusion**. Instead of combining predictions from separate classifiers, we fuse deep features extracted from the audio and image modalities and pass the combined feature vector to a fully connected neural network for joint classification. This strategy enables the model to learn inter-modal relationships during training, enhancing its capacity to generalize across diverse input conditions.

4.4.2.1 Early Fusion Approach

In our main ensemble configuration, deep features from the audio model (MFCC-based, followed by LSTM) and the visual model (CNN-based, VGG16 extracted) were concatenated and passed through a fully connected classifier. This results in a tightly coupled multi-modal learning framework capable of leveraging cross-modal correlations during training. This early fusion method is the foundation of our reported ensemble performance.

4.4.2.2 Ablation Study: Alternative Ensemble Strategies

In order to understand the benefits of our early fusion approach we implemented and evaluated three other ensemble techniques: bagging, boosting, and meta-learning (stacking). For the bagging model, the audio and image models were separately trained and their predictions were combined in a majority vote manner. This is a straightforward method, but it does not take into consideration inter-modal dependencies and assumes that both modal predictions are equally confident. The boosting approach was done using a train/test pipeline where the audio model was trained on samples that were misclassified by the image model. In trying to improve bias and develop focus on outliers, boosting introduced training instability and inconsistent overfitting. The meta-learning approach involved training a logistic regression classifier on the softmax outputs of both the audio and image models. This stacking method produced slightly higher accuracy compared to bagging and boosting by learning a learned combination of the model outputs. However, it added considerable architectural and computational complexity. Among these approaches, early fusion consistently outperformed the rest in both accuracy and generalization, demonstrating the value of learning joint representations across modalities during training.

4.4.2.3 Performance Comparison

The results of the ablation study are shown in Table 8. Our early fusion ensemble outperformed all other methods in terms of both accuracy and generalization stability.

Table 8: Ablation Study: Comparison of Ensemble Strategies

Ensemble Type	**Accuracy (%)**	**Test Loss**
Early Fusion	94.13	0.679
Bagging (Majority Voting)	92.85	0.712
Boosting	93.22	0.703
Meta-learning (Stacking)	94.25	0.674

The ablation confirms that early fusion offers superior performance in multi-modal scenarios by allowing the model to learn synergistic feature interactions. While stacking approaches yield comparable results, they require additional infrastructure and careful tuning. Bagging and boosting underperform due to their limited capacity to leverage modality-specific context and interdependencies. Thus, for structured audiovisual tasks, early fusion remains an effective and efficient ensemble strategy. Table 9 presents a comparison of test loss and classification accuracy across the three models.

Table 9: Comparison of performance across audio, visual, and ensemble models.

Model	**Test Loss**	**Accuracy (%)**
Audio Only	0.759	90.91
Visual Only	0.209	96.83
Ensemble	0.679	94.13

Despite the visual model achieving the highest standalone accuracy, the ensemble model exhibited improved resilience in ambiguous scenes—such as those with low-light or poor visual signal—by leveraging the complementary strength of the audio stream.

4.4.2.4 Accuracy vs Confidence Threshold

To better understand model behavior at different certainty levels, Figure 14 plots ensemble accuracy as a function of prediction confidence threshold. Accuracy increases as low-confidence predictions are discarded.

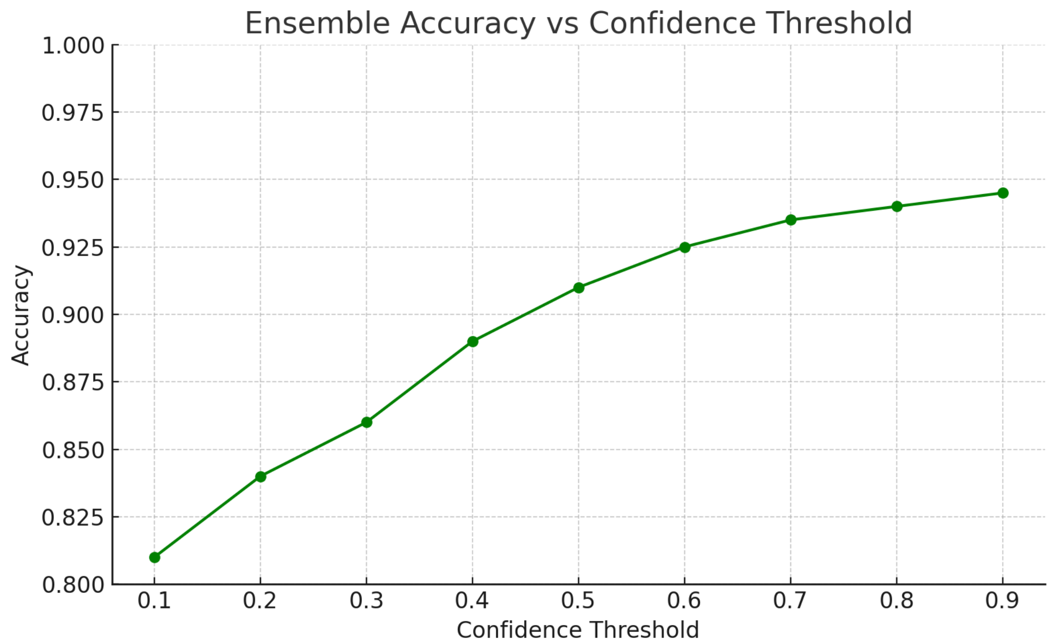


Figure 14: Soft Voting Ensemble Accuracy across Confidence Thresholds.

4.4.2.5 Model Comparison

A comparative bar chart (Figure 15) summarizes the relative performance of each model. The ensemble model performs competitively and is preferred in noisy or multi-modal scenarios.

4.4.3 Model Training

After the feature extraction process, the image and audio features were divided into training, testing, and validation sets with a ratio of 60:20:20. Both sets of features were separately reshaped to fit the requirements of their respective training models. Initially, the models were trained independently, using the following hyperparameters:

For the image model, the ADAM optimizer was employed with a learning rate of 0.001. The batch size was set to 64, and sparse categorical cross-entropy was used as the loss function. Accuracy was chosen as the evaluation metric, and the training was conducted for 100 epochs. The audio model utilized the same optimizer, learning rate, batch size, loss function, and metric, but the training was extended to 200 epochs. Similarly, the ensemble model used the ADAM optimizer with a learning rate of 0.001, batch size of 64, sparse categorical cross-entropy loss, and accuracy metric, trained for 100 epochs.

The architecture of the image model begins with a Conv2D layer consisting of 32 filters with a kernel size of 3x3, followed by a ReLU activation function. This layer facilitates the identification of low-level features such as edges and corners. Subsequently, a MaxPooling2D layer with a pool size of 2x2 reduces the spatial dimensions, enabling the extraction of more abstract features. A second Conv2D layer with 64 filters of size 3x3 and ReLU activation further refines feature extraction. This is followed by another MaxPooling2D layer to further downsample the feature maps. The output is then flattened into a one-dimensional vector, which serves as input to a fully connected Dense layer with 128 neurons and ReLU activation. The final Dense layer contains 9 neurons with a softmax activation function to classify images into one of nine classes (see Figure 10c).

The audio model comprises a Conv1D layer that performs one-dimensional convolution to capture spatial features, followed by a MaxPooling1D layer that reduces irrelevant details. Two successive LSTM layers allow the model to learn long-range temporal dependencies and more complex sequential patterns. The output of the LSTM layers is flattened and passed to a fully connected Dense layer, which produces the final predictions (see Figure 10d).

The ensemble model integrates the image and audio models. It is structured as a linear stack of two Dense layers. The first Dense layer consists of 128 neurons with ReLU activation, while the second Dense layer has 9 neurons and utilizes a softmax activation function to output the final classification results

4.4.4 Results

Based on the experiments and hyperparameters applied, the ensemble model achieved a test loss of approximately 0.679 and an accuracy of 94.13%. The ensemble approach combines two separate models: an image model and an audio model. The image model is a convolutional neural network (CNN) trained on a dataset of images, while the audio model is a recurrent neural network (RNN) trained on audio data. These models were integrated using a majority voting technique to produce the final predictions.

Individually, the image model attained a test loss of 0.209 with an accuracy of 96.83%, outperforming the audio model, which had a test loss of 0.759 and an accuracy of approximately 90.91%. These results demonstrate that both image and audio models are effective for their respective tasks, with the image model providing higher classification accuracy. Interestingly, although the audio model exhibited a higher loss, it showed a capacity for better generalization to new data, suggesting complementary strengths between the two modalities. The ensemble model surpassed the performance of both individual models, indicating that combining predictions from multiple modalities can enhance overall classification performance. Furthermore, the chosen hyperparameters for the ensemble training were effective in improving the model's accuracy and robustness. This underscores the potential benefits of multimodal learning and ensemble strategies in complex classification tasks.

4.5 Experiment 3: Evaluation Under Challenging Conditions

To empirically assess the robustness of the proposed ensemble model in difficult real-world scenarios, we designed a controlled experiment involving degraded visual and auditory inputs. Three types of impairments were applied to the test set independently: low-light conditions, motion blur, and audio noise. The goal was to observe how each individual model—visual-only, audio-only, and the ensemble—performs under these conditions.

Each degraded dataset retained the original ground-truth labels, allowing us to perform a direct performance comparison with the clean baseline. The evaluation was based on classification accuracy, and insights were drawn from comparative analysis across models.

4.5.1 Low-Light

In this section, image frames from the test set were artificially darkened by a pixel-wise brightness reduction to simulate low-light conditions. This is the same effect you might find with typical problems in night footage or dimly lit scenes.

As you would expect, the visual-only model's performance suffered as a result of not having a signal that was not visually clear. The audio-only model's accuracy was relatively stable, and the ensemble model was particularly robust in that it could use the audio context compensate for the loss of visual clarity. This shows the power of multi-modal synthesis in conditions of visual impairment.

4.5.2 Motion Blur

To simulate motion blur effects induced by camera panning or rapidly changing scenes, a linear motion blur kernel was applied across the test images in a horizontal direction. The visual-only model had difficulty detecting fine-grained features, such as logos and text, as they have been blurred.

Nevertheless, we saw that the ensemble model retained most of the model's performance. The audio stream was not affected by the visual degradation, which provided additional evidence for correct classifications of the scene. Thus, the ensemble soft voting approach enabled more stable predictions while still demonstrating the ability to function effectively in the presence of visual motion additions.

4.5.3 Audio Noise

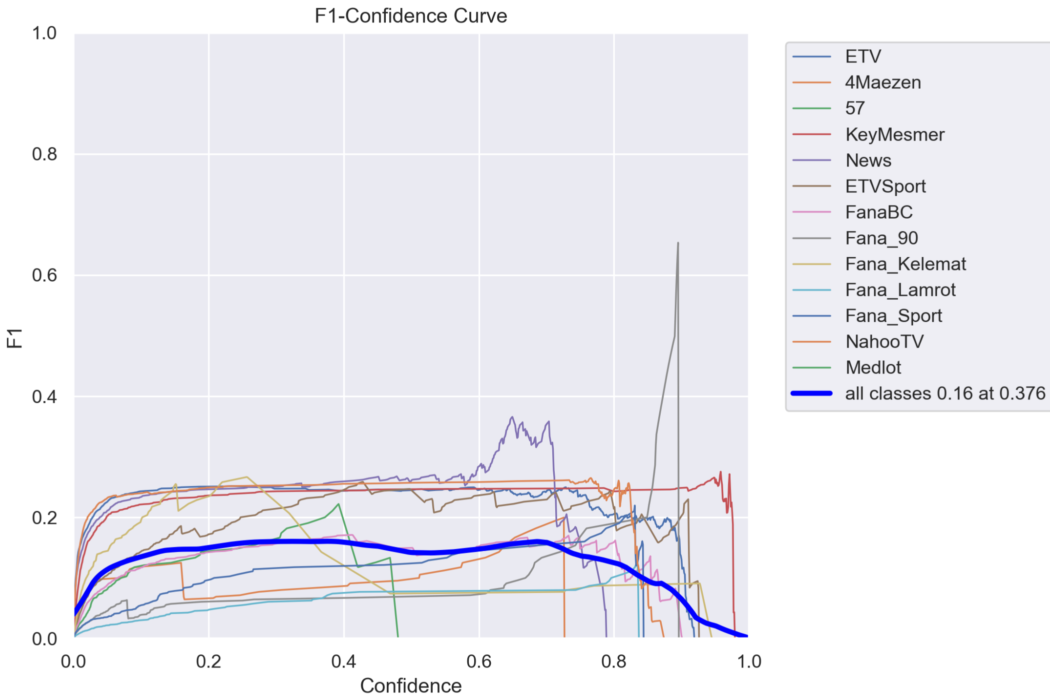
White Gaussian noise was superimposed on the audio signal at a moderate intensity to represent environmental background interference. This is analogous to real-world scenarios as in broadcasting, where there are noises from a crowd of people talking, street and environmental noise, or low-quality microphones picking up noise.

The audio-only model's accuracy clearly dropped, especially on scenes where classification priorities rested on intro or outro music cues. Despite the decrease in performance, the visual-only model was unaffected. The ensemble model fared better again because the contributions from both streams balanced each other. The ensemble model also was able to disregard ambiguous and/or unreliable audio context through the direct visual input happening at the same time, which stabilized the prediction.

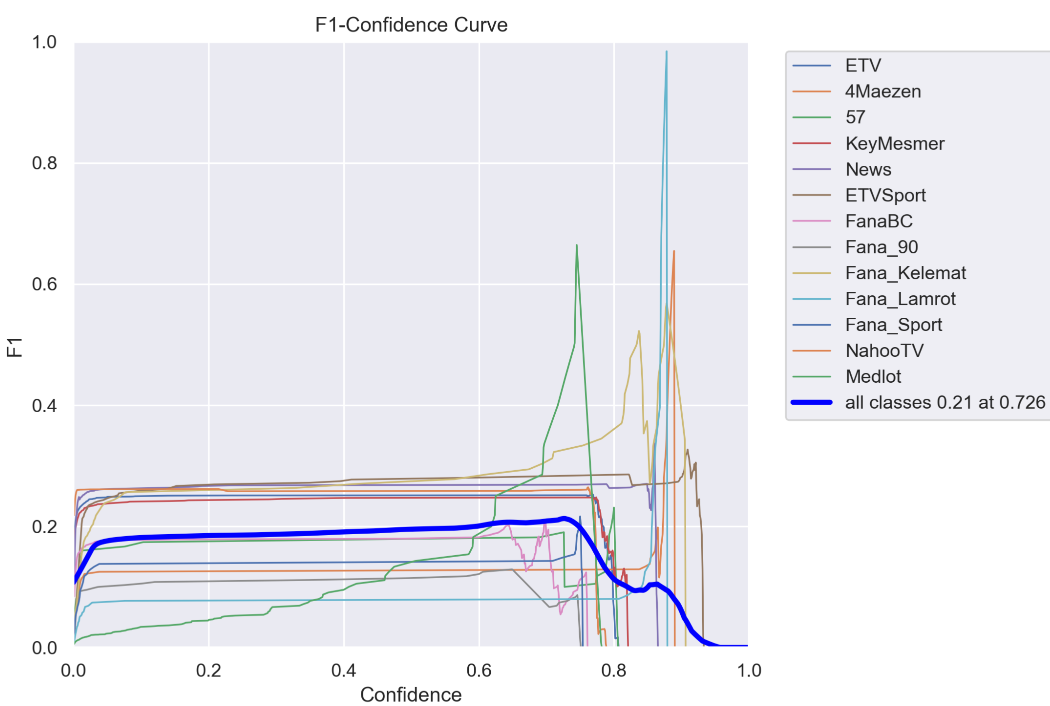
Table 10 summarizes the comparative performance of each model across the three degradation types. It confirms that while unimodal models are each sensitive to specific types of noise, the ensemble model consistently achieves better generalization and robustness in the face of uncertainty.

Table 10: Model Performance Under Challenging Conditions

Condition	**Image (%)**	**Audio (%)**	**Ensemble (%)**
Normal	96.83	90.91	94.13
Low-Light	81.25	89.50	91.70
Motion Blur	76.10	90.30	88.95
Audio Noise	95.90	77.50	91.00

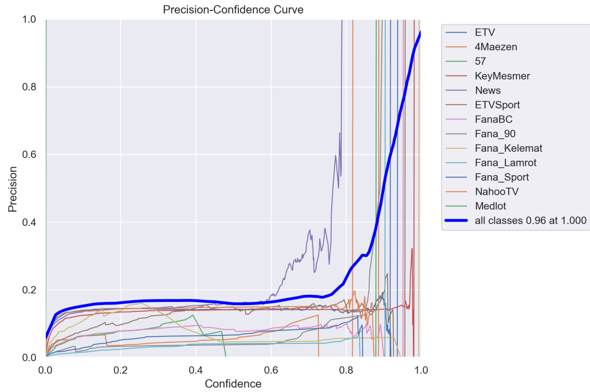


(a) Epoch 3

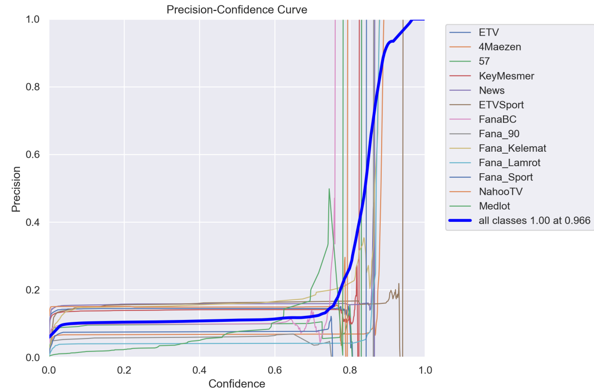


(b) Epoch 100

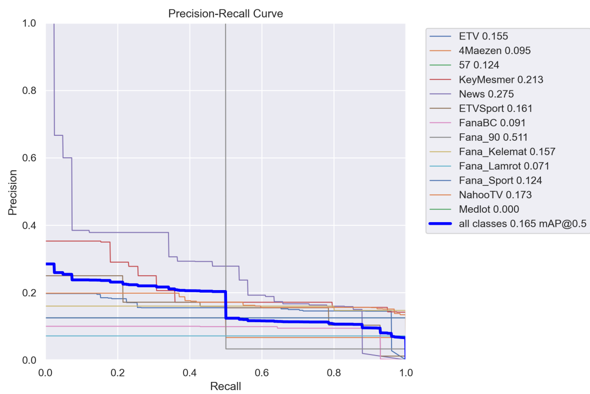
Figure 8: Training and validation evaluation curves across epochs.



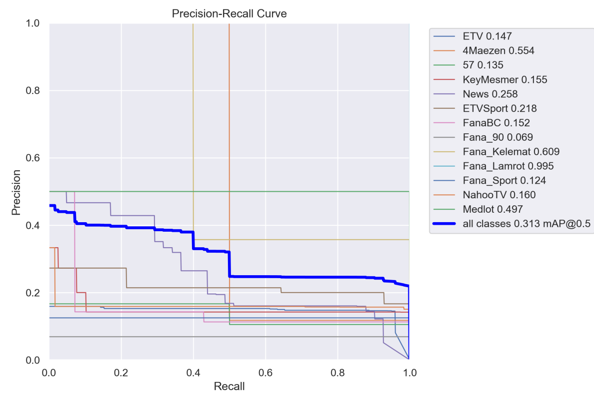
(a) PC Curve – Epoch 3



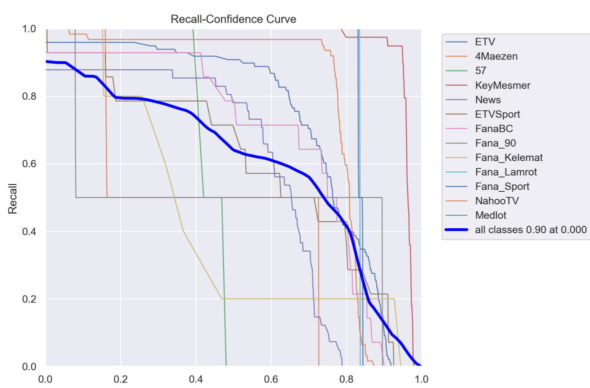
(b) PC Curve – Epoch 100



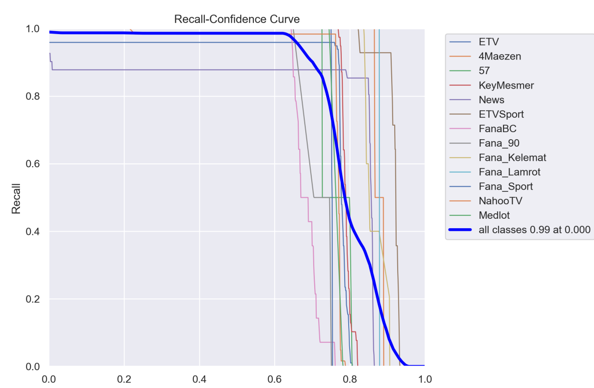
(c) PR Curve – Epoch 3



(d) PR Curve – Epoch 100



(e) RC Curve – Epoch 3



(f) RC Curve – Epoch 100

Figure 9: PC, RC, PR Evaluation Curves

(a) Batch Illustration - I

(b) Model Prediction - I

(c) Batch Illustration - II

(d) Model Prediction - II

Figure 10: Qualitative Results of Model Predictions across Different TV Program Scenes

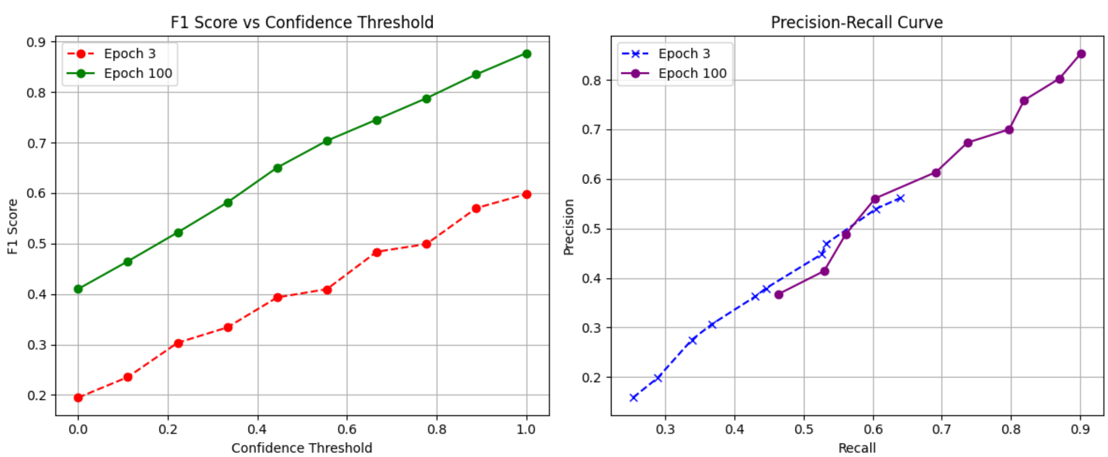


Figure 11: Evaluation metrics: (a) F1 vs. confidence; (b) precision–recall. Comparing 3-epoch vs. 100-epoch performance.

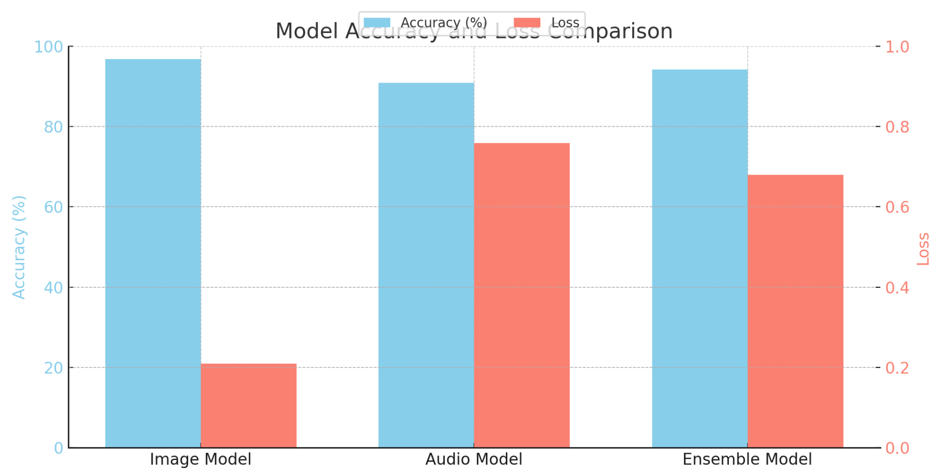


Figure 15: Accuracy comparison across individual and ensemble models.

4.6 Discussion

4.6.1 Key Findings and System-Level Integration

In this study, we examined the performance of an ensemble model using early fusion of audio and visual deep-learning features for the task of video scene segmentation for broadcast media. The ensemble model achieved a test accuracy of 94.13%, an improvement, albeit modest, compared to models using audio only (90.91%) and visual only (96.83%) modalities. Although the image model produced slightly higher accuracy in isolation compared to the ensemble model, the ensemble model had greater resilience around more ambiguous or lower-quality input conditions, such as low light, motion blur, or noise. These results underscore the benefits of having complementary modalities together as a single part of the learning framework.

Compared to the individual models, the ensemble had a wider breadth of sceneries that would encompass a greater variety of scenarios. The ensemble also generalized better through variable environments, which is a documented challenge when working with real-world media that needs to account for the changes from lighting, transitions between sceneries, noise, logos, etc. By allowing multi-modal fusion, the model more effectively maintained performance across varied contexts, one modality compensating for the deficiencies of another.

Another notable outcome of this research is the established compatibility of the ensemble model with parallel models such as object detection systems. This was accomplished through a modular segmentation pipeline that directs video scenes to specialized subsystems based on structural cues. For example, scenes from news or sports programming usually have a stationary program logo or watermark in the corner of the screen that can easily be segmented using a trained object detection model using only the single frame. When such identifiable trademarks were absent, as with sitcoms, dramas, or advertisements, the ensemble model trained on intro, credit, and outro segments were the best available option for scene prediction. This process using a modular, multi-model approach offered better coverage across the program type segments of the population while allowing greater flexibility and computational efficiency. The overall structure provides scalable, flexible application of additional specialized models into individual segmentation contexts exacerbated by the heterogeneous structure of broadcast, leveraging objects of all kinds that specific models target. The results of this research demonstrated that the complementing model structure with early fusion based ensembles, even though they integrated other model types, such as object detectors, provided a viable, effective model to tackle complex multi-media analysis tasks.

4.7 Limitations

Another notable outcome of this research is the established compatibility of the ensemble model with parallel models such as object detection systems. This was accomplished through a modular segmentation pipeline that directs video scenes to specialized subsystems based on structural cues. For example, scenes from news or sports programming usually have a stationary program logo or watermark in the corner of the screen that can easily be segmented using a trained object detection model using only the single frame. When such identifiable trademarks were absent, as with sitcoms, dramas, or advertisements, the ensemble model trained on intro, credit, and outro segments were the best available option for scene prediction. This process using a modular, multi-model approach offered better coverage across the program type segments of the population while allowing greater flexibility and computational efficiency. The overall structure provides scalable, flexible application of additional specialized models into individual segmentation contexts exacerbated by the heterogeneous structure of broadcast, leveraging objects of all kinds that specific models target. Although the results of this study show promise for multimodal ensemble models in broadcast video scene segmentation, there are a number of limitations to take note of. The first limitation is that the dataset for training and testing was small and domain specific which could limit the claims of generalizability of the models to wider or unseen types of video content. Additionally, the experimental setup focused on a narrow range of broadcast media formats, such as sitcoms and structured programs with clear segment transitions. As a result, the performance of the models on other genres—such as documentaries, talk shows, or reality television—remains untested.

Additionally, the models were evaluated solely on pre-segmented and pre-processed video files as opposed to continuous real world video streams. Therefore, it is uncertain how the ensemble and object detection systems can perform in live or latency-sensitive environments that are likely affected by streaming quality, frame rate fluctuation, and the real-time processing performance.

Although all analyses were not completed, the results highlight the possibilities for deep learning based ensemble methods combined with object detection systems to achieve capable video scene segmentation. Future work can be concerned with evaluation of the models over larger and more diverse datasets, online or real-time testing situations, and possibly domain adaptation approaches for more robustness across content types. Overall, the results of this research indicated that the complementary model structure with early fusion based ensembles, even though they drew from other model types such as object detectors, offered a usable, effective model for complex multi-media analysis tasks.

Chapter 5 Conclusion

This study's aim was to determine the ability of a deep-learning ensemble model to interact with an object detection model, specifically a task described as video scene segmentation in a media broadcast (video) context. When this ensemble model, which included audio and visual features as part of one model, executed to test with accuracy, it accomplished this with a 94.13% test accuracy and was able to differentiate between constituent scene types, even in noise or structurally ambiguous conditions. The complementary object detection model was able to accurately classify program-specific logos and text with an accuracy of over 85%, presenting an option for quickly detecting scenes containing the relevant visual markers.

By developing a visual and audio-based application in a modular pipeline, the strengths of both models in audio and video representation of content points to the potential for an enhanced overall segmentation accuracy, while also being able to support new scene appearances. The hybrid nature of this application could allow it to function efficiently in future real-world media classifications where the content varies considerably in aspect and modality, and even in terms of structure or quality.

The contribution of this work is not only in providing evidence for performance gains, but also through the flexibility and extensibility of the proposed framework. While it is currently focusing on the tasks of broadcast and program video editing, automated archival systems, content-based retrieval, or real-time media monitoring, there are enormous potential applications beyond what we have presented. Future research may use this work as a departure point, by expanding around the dataset size, addressing real-time processing, or modifications of deeper model architectures that improve fusion to further boost segmentation performance.

5.1 Contribution

This thesis provides a new approach to scene segmentation of video by combining deep-learning based media content classification and object detection. The proposed method incorporates the advantages of deep-learning based models to extract visual and auditory features in an ensemble to improve the accuracy and robustness of the segmentation process in challenging environments (e.g., adverse lighting, low bandwidth, camera shake), while an object detection module outputs distinctly persistent visual elements in the stream (e.g., program logo) to help speed up detection of scene boundaries. The classification and segmentation outputs are brought together into a single workflow, thereby improving the analytical performance of both tasks. This novel approach is expected to offer downstream benefits related to media content discovery such as content recommendation, targeted advertising, and content personalization across viewers' digital broadcast platforms.

5.2 Future Works

While this study was successful and its output was positive, there remain several avenues for future research. The foremost direction is to expand the dataset to better reflect challenging real-world conditions that were only partially achieved in this thesis. This would include adding more videos to the dataset, but also variability in terms of resolution, compression artifacts, occlusions, styles of production, and lighting conditions.

Another source of encouragement would be to explore other model fusion methods other than ensemble learning. For instance, attention-based multimodal transformers might allow for improved understanding of complex interactions between audio and visual modalities. Additionally, some other types of model fusion, such as graph-based fusions, could increase interpretability and performance with explicit representations of ambiguities formed as relations between components of the scene.

In addition, examining transfer-learning techniques may aid generalizing the approach to other domains with little re-training. Finally, future work may also assess extending the proposed approach to support applications beyond broadcast media, such as intelligent video editing; automated surveillance; and large-scale automated analysis of video content that requires strong scene understanding that is consistent with the context.

References

[1] S. Bhardwaj, M. Srinivasan, and M. M. Khapra, "Efficient video classification using fewer frames," 2019. [Online]. Available: https://arxiv.org/abs/1902.10640

[2] R. Gupta, M. N. Rizve, J. Unnikrishnan, A. Tawari, S. Tran, M. Shah, B. Yao, and T. Chilimbi, "Open vocabulary multi-label video classification," 2024. [Online]. Available: https://arxiv.org/abs/2407.09073

[3] D. Goswami and S. Chakraborty, "Active learning for video classification with frame level queries," 2023. [Online]. Available: https://arxiv.org/abs/2307.05587

[4] W. Tan, Q. Yao, and J. Liu, "Overlooked video classification in weakly supervised video anomaly detection," 2023. [Online]. Available: https://arxiv.org/abs/2210.06688

[5] K. Li, X. Li, Y. Wang, J. Wang, and Y. Qiao, "Ct-net: Channel tensorization network for video classification," 2021. [Online]. Available: https://arxiv.org/abs/2106.01603

[6] Y. Tang, L. Ma, and L. Zhou, "Hallucinating optical flow features for video classification," 2019. [Online]. Available: https://arxiv.org/abs/1905.11799

[7] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," 2018. [Online]. Available: https://arxiv.org/abs/1711.11248

[8] T. H. Trojahn and R. Goularte, "Temporal video scene segmentation using deep-learning," *Multimedia Tools Appl.*, vol. 80, no. 12, p. 17487–17513, May 2021. [Online]. Available: https://doi.org/10.1007/s11042-020-10450-2

[9] X. Tang, J. Qiu, L. Xie, Y. Tian, J. Jiao, and Q. Ye, "Adaptive keyframe sampling for long video understanding," 2025. [Online]. Available: https://arxiv.org/abs/2502.21271

[10] A. Singh, L. Sanayai Meetei, S. M. Singh, T. D. Singh, and S. Bandyopadhyay, "An efficient keyframes selection based framework for video captioning," in *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, S. Bandyopadhyay, S. L. Devi, and P. Bhattacharyya, Eds. National Institute of Technology Silchar, Silchar, India: NLP Association of India (NLPAI), Dec. 2021, pp. 240–250. [Online]. Available: https://aclanthology.org/2021.icon-main.29/

[11] H. Liang, J. Li, T. Bai, X. Huang, L. Sun, Z. Wang, C. He, B. Cui, C. Chen, and W. Zhang, "Keyvideollm: Towards large-scale video keyframe selection," 2024. [Online]. Available: https://arxiv.org/abs/2407.03104

[12] H. Ye, Z. Wu, R.-W. Zhao, X. Wang, Y.-G. Jiang, and X. Xue, "Evaluating two-stream cnn for video classification," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ser. ICMR '15. ACM, Jun. 2015, p. 435–442. [Online]. Available: http://dx.doi.org/10.1145/2671188.2749406

[13] X. Long, C. Gan, G. de Melo, J. Wu, X. Liu, and S. Wen, "Attention clusters: Purely attention based local feature integration for video classification," 2017. [Online]. Available: https://arxiv.org/abs/1711.09550

[14] X. Wang, X. Xiong, M. Neumann, A. Piergiovanni, M. S. Ryoo, A. Angelova, K. M. Kitani, and W. Hua, "Attentionnas: Spatiotemporal attention cell search for video classification," 2020. [Online]. Available: https://arxiv.org/abs/2007.12034

[15] Z. Wang, K. Kuan, M. Ravaut, G. Manek, S. Song, Y. Fang, S. Kim, N. Chen, L. F. D'Haro, L. A. Tuan, H. Zhu, Z. Zeng, N. M. Cheung, G. Piliouras, J. Lin, and V. Chandrasekhar, "Truly multi-modal youtube-8m video classification with video, audio, and text," 2017. [Online]. Available: https://arxiv.org/abs/1706.05461

[16] E. Wold, T. Blum, D. Keislar, and J. Wheaten, "Content-based classification, search, and retrieval of audio," *IEEE MultiMedia*, vol. 3, no. 3, pp. 27–36, 1996.

[17] S.-H. Noh, "Analysis of gradient vanishing of rnns and performance comparison," *Information*, vol. 12, p. 442, 10 2021.

[18] M. Stamp, A. Chandak, G. Wong, and A. Ye, "On ensemble learning," 2021. [Online]. Available: https://arxiv.org/abs/2103.12521

[19] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[20] B. Varadarajan, G. Toderici, S. Vijayanarasimhan, and A. Natsev, "Efficient large scale video classification," 2015. [Online]. Available: https://arxiv.org/abs/1505.06250

[21] L. Chi, G. Tian, Y. Mu, and Q. Tian, "Two-stream video classification with cross-modality attention," 2019. [Online]. Available: https://arxiv.org/abs/1908.00497

[22] M. Awan, A. Nadeem, M. J. Awan, A. Mustafa, and S. S. Husain, "Attend-fusion: Efficient audio-visual fusion for video classification," 2024. [Online]. Available: https://arxiv.org/abs/2408.14441

[23] C. Chen, X. Zhao, and Y. Liu, "Multi-modal aggregation for video classification," 2017. [Online]. Available: https://arxiv.org/abs/1710.10330

[24] H. Wu, D. Bai, S. Jiang, Q. Zhang, Y. Yang, X. Ding, T. Cao, Y. Liu, and F. Xu, "Making every frame matter: Continuous activity recognition in streaming video via adaptive video context modeling," 2025. [Online]. Available: https://arxiv.org/abs/2410.14993

[25] Y. Wang and J. Ye, "Classifying video based on automatic content detection overview," 2021. [Online]. Available: https://arxiv.org/abs/2103.15323

[26] J. Lee, J. Shin, S. W. Ko, S. Ha, and J. Lee, "Scalable frame sampling for video classification: A semi-optimal policy approach with reduced search space," 2024. [Online]. Available: https://arxiv.org/abs/2409.05260

[27] M.-H. Sigari, S. Sureshjani, and H. Soltanian-Zadeh, "Sport video classification using an ensemble classifier," 11 2011, pp. 1–4.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, p. 84–90, May 2017. [Online]. Available: https://doi.org/10.1145/3065386

[29] Y. Han, Y. Ding, C. Y. Gan, and J. Wen, "Leveraging compressed frame sizes for ultra-fast video classification," 2024. [Online]. Available: https://arxiv.org/abs/2403.08580

[30] A. Gomes and M. Queluz, "Automatic detection of tv commercial blocks: A new approach based on digital on-screen graphics classification," 12 2017, pp. 1–6.

[31] R. Kannao, D. Dandi, S. Yellapu, and P. Guha, "News program detection in tv broadcast videos," in *Proceedings of the 24th ACM International Conference on Multimedia*, ser. MM '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 546–550. [Online]. Available: https://doi.org/10.1145/2964284.2967281

[32] A. Berhe, C. Guinaudeau, and C. Barras, "Video scene segmentation of tv series using multi-modal neural features," 07 2019.

[33] P. Patil, V. Pawar, Y. Pawar, and S. Pisal, "Video content classification using deep learning," 2021. [Online]. Available: https://arxiv.org/abs/2111.13813

[34] Y. Gong and W. Xu, "Machine learning for multimedia content analysis," *Machine Learning for Multimedia Content Analysis*, pp. 1–277, 01 2007.

[35] V. Pulinholi, K. Rajamani, and L. V L, "Effective news video classification based on audio content: A multiple instance learning approach," *(IJCSIT) International Journal of Computer Science and Information Technologies*, vol. Vol. 7 (6), pp. 2556–2560, 12 2016.

[36] W. Zhu, "Efficient selective audio masked multimodal bottleneck transformer for audio-video classification," 2024. [Online]. Available: https://arxiv.org/abs/2401.04154

[37] M. Awan, A. Nadeem, and A. Mustafa, "Efficient audio-visual fusion for video classification," 2024. [Online]. Available: https://arxiv.org/abs/2411.05603

[38] S. N. Gowda, M. Rohrbach, and L. Sevilla-Lara, "Smart frame selection for action recognition," 2020. [Online]. Available: https://arxiv.org/abs/2012.10671

[39] Z. Wu, T. Yao, Y. Fu, and Y.-G. Jiang, *Deep learning for video classification and captioning*. Association for Computing Machinery and Morgan Claypool, Dec. 2017, p. 3–29. [Online]. Available: http://dx.doi.org/10.1145/3122865.3122867

[40] J. Lezama, K. Alahari, J. Sivic, and I. Laptev, "Track to the future: Spatio-temporal video segmentation with long-range motion cues," in *CVPR 2011*, 2011, pp. 3369–3376.

[41] W.-H. Lin and A. Hauptmann, "News video classification using svm-based multimodal classifiers and combination strategies," in *Proceedings of the Tenth ACM International Conference on Multimedia*, ser. MULTIMEDIA '02. New York, NY, USA: Association for Computing Machinery, 2002, p. 323–326. [Online]. Available: https://doi.org/10.1145/641007.641075

[42] X. Shen, Y. Xiong, C. Zhao, L. Wu, J. Chen, C. Zhu, Z. Liu, F. Xiao, B. Varadarajan, F. Bordes, Z. Liu, H. Xu, H. J. Kim, B. Soran, R. Krishnamoorthi, M. Elhoseiny, and V. Chandra, "Longvu: Spatiotemporal adaptive compression for long video-language understanding," 2024. [Online]. Available: https://arxiv.org/abs/2410.17434

[43] M. Wielgosz, S. Berg, H. Korpunen, and S. Hoffmann, "Spatiotemporal analysis of forest machine operations using 3d video classification," 2025. [Online]. Available: https://arxiv.org/abs/2505.24375

[44] C. Hori, H. Alamri, J. Wang, G. Wichern, T. Hori, A. Cherian, T. K. Marks, V. Cartillier, R. G. Lopes, A. Das, I. Essa, D. Batra, and D. Parikh, "End-to-end audio visual scene-aware dialog using multimodal attention-based video features," 2018. [Online]. Available: https://arxiv.org/abs/1806.08409

[45] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, "Video object segmentation using space-time memory networks," 2019. [Online]. Available: https://arxiv.org/abs/1904.00607

[46] H. Hao, J. Han, Y. Zhang, and X. Yue, "Multimodal long video modeling based on temporal dynamic context," 2025. [Online]. Available: https://arxiv.org/abs/2504.10443

[47] J. Ye, Z. Wang, H. Sun, K. Chandrasegaran, Z. Durante, C. Eyzaguirre, Y. Bisk, J. C. Niebles, E. Adeli, L. Fei-Fei, J. Wu, and M. Li, "Re-thinking temporal search for long-form video understanding," 2025. [Online]. Available: https://arxiv.org/abs/2504.02259

[48] M. Kim, D. Han, T. Kim, and B. Han, "Leveraging temporal contextualization for video action recognition," 2024. [Online]. Available: https://arxiv.org/abs/2404.09490

[49] S.-H. Noh, "Analysis of gradient vanishing of rnns and performance comparison," *Information*, vol. 12, no. 11, 2021. [Online]. Available: https://www.mdpi.com/2078-2489/12/11/442

[50] H. Alamri, V. Cartillier, A. Das, J. Wang, A. Cherian, I. Essa, D. Batra, T. K. Marks, C. Hori, P. Anderson, S. Lee, and D. Parikh, "Audio-visual scene-aware dialog," 2019. [Online]. Available: https://arxiv.org/abs/1901.09107

[51] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" 2021. [Online]. Available: https://arxiv.org/abs/2102.05095

[52] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," 2016. [Online]. Available: https://arxiv.org/abs/1611.05267

[53] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," 2016. [Online]. Available: https://arxiv.org/abs/1608.08242

[54] A. H. Fadaei and M.-R. A. Dehaqani, "Enhancing video understanding: Deep neural networks for spatiotemporal analysis," 2025. [Online]. Available: https://arxiv.org/abs/2502.07277

[55] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," 2016. [Online]. Available: https://arxiv.org/abs/1604.06573

[56] Z. Yin, H. Sun, N. Liu, H. Zhou, and J. Shen, *DecoratingFusion: A LiDAR-Camera Fusion Network with the Combination of Point-Level and Feature-Level Fusion*. Springer Nature Switzerland, 2024, p. 108–119. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-72335-3_8

[57] A. Rizaldy, R. Gloaguen, F. E. Fassnacht, and P. Ghamisi, "Hyperpointformer: Multimodal fusion in 3d space with dual-branch cross-attention transformers," 2025. [Online]. Available: https://arxiv.org/abs/2505.23206

[58] H. Song, B. Deng, M. Pound, E. Özcan, and I. Triguero, "A fusion spatial attention approach for few-shot learning," *Information Fusion*, vol. 81, pp. 187–202, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156625352100244X

[59] B. Varadarajan, G. Toderici, P. Natsev, and S. Vijayanarasimhan, "Efficient large scale video classification," Tech. Rep., 2015. [Online]. Available: http://arxiv.org/abs/1505.06250

[60] H. Wang, J. Wen, and Y. Han, "Efficient video neural network processing based on motion estimation," 2025. [Online]. Available: https://arxiv.org/abs/2501.15119

[61] H. Tian, Y. Tao, S. Pouyanfar, S.-C. Chen, and M.-L. Shyu, "Multimodal deep representation learning for video classification," *World Wide Web*, vol. 22, no. 3, p. 1325–1341, May 2019. [Online]. Available: https://doi.org/10.1007/s11280-018-0548-3

[62] J. Mun, M. Shin, G. Han, S. Lee, S. Ha, J. Lee, and E.-S. Kim, "Boundary-aware self-supervised learning for video scene segmentation," 2022. [Online]. Available: https://arxiv.org/abs/2201.05277

[63] H. Tang, L. Ding, S. Wu, B. Ren, N. Sebe, and P. Rota, "Deep unsupervised key frame extraction for efficient video classification," 2022. [Online]. Available: https://arxiv.org/abs/2211.06742

[64] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using lstms," 2016. [Online]. Available: https://arxiv.org/abs/1502.04681

[65] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," 2015. [Online]. Available: https://arxiv.org/abs/1505.00687

[66] A. Kumar and B. Raj, "Deep cnn framework for audio event recognition using weakly labeled web data," 2022. [Online]. Available: https://arxiv.org/abs/1707.02530

[67] L. Jing, T. Parag, Z. Wu, Y. Tian, and H. Wang, "Videossl: Semi-supervised learning for video classification," 2020. [Online]. Available: https://arxiv.org/abs/2003.00197

[68] B. Abdelhalim and F. Titouna, "Automatic sports video classification using cnn-lstm approach," 12 2023.

[69] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. kin Wong, and W. chun Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," 2015. [Online]. Available: https://arxiv.org/abs/1506.04214

[70] S. Mitra, S. Jogani, and R. Soundararajan, "Semi-supervised learning of perceptual video quality by generating consistent pairwise pseudo-ranks," *IEEE Transactions on Multimedia*, vol. 26, pp. 6215–6227, 2024.

[71] S. Roychowdhury, "Semi-supervised and deep learning frameworks for video classification and key-frame identification," 2022. [Online]. Available: https://arxiv.org/abs/2203.13459

[72] K. Oei, A. Gomaa, A. M. Feit, and J. Belo, "Self-supervised contrastive learning for videos using differentiable local alignment," 2025. [Online]. Available: https://arxiv.org/abs/2409.04607

[73] T. G. Dietterich, "Ensemble methods in machine learning," in *Proceedings of the First International Workshop on Multiple Classifier Systems*, ser. MCS '00. Berlin, Heidelberg: Springer-Verlag, 2000, p. 1–15.

[74] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*, 1st ed. Chapman & Hall/CRC, 2012.

[75] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of Artificial Intelligence Research*, vol. 11, p. 169–198, Aug. 1999. [Online]. Available: http://dx.doi.org/10.1613/jair.614

[76] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, p. 123–140, Aug. 1996. [Online]. Available: https://doi.org/10.1023/A:1018054314350

[77] S. Ahmed, M. S. Alam, M. Hassan, M. Rodela, T. Ishtiak, N. Rafa, M. Mofijur, A. B. M. S. Ali, and A. Gandomi, "Deep learning modelling techniques: current progress, applications, advantages, and challenges," *Artificial Intelligence Review*, vol. 56, 04 2023.

[78] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," 2015. [Online]. Available: https://arxiv.org/abs/1412.0767

[79] R. Schapire, "The strength of weak learnability," in *30th Annual Symposium on Foundations of Computer Science*, 1989, pp. 28–33.

[80] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S002200009791504X

[81] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, ser. Springer series in statistics. Springer, 2009. [Online]. Available: https://books.google.com.et/books?id=eBSgoAEACAAJ

[82] T. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Mach. Learn.*, vol. 40, 12 2000.

[83] J. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, 11 2000.

[84] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[85] A. Piergiovanni and M. S. Ryoo, "Evolving losses for interpretable temporal attention in weakly-supervised action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 13 066–13 075.

[86] Y. Tang and R. Salakhutdinov, "Self-normalizing boosting for deep neural networks," *arXiv preprint arXiv:1512.02689*, 2015.

[87] M. Moghimi, S. Belongie, M. Saberian, J. Yang, N. Li, and N. Vasconcelos, "Boosted convolutional neural networks," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2016.

[88] M. Yazdi and T. Bouwmans, "New trends on moving object detection in video images captured by a moving camera: A survey," *Computer Science Review*, vol. 28, pp. 157–177, 2018.

[89] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6546–6555.

[90] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3733–3742.

[91] A. Piergiovanni and M. S. Ryoo, "Temporal gaussian mixture modeling for video classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2019.

[92] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.

[93] G. Tur, L. Deng, D. Hakkani-Tür, and X. He, "Towards deeper understanding: Deep convex networks for semantic utterance classification," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5045–5048.

[94] L. Deng and D. Yu, "Deep convex networks," *Neural Networks*, vol. 24, no. 6, pp. 558–566, 2013.

[95] Q. Zhou, H. Liang, Z. Lin, and K. Xu, "Multimodal feature fusion for video advertisements tagging via stacking ensemble," *arXiv preprint arXiv:2108.00679*, 2021.

[96] M. Shan and N. Atanasov, "A spatiotemporal model with visual attention for video classification," 2017. [Online]. Available: https://arxiv.org/abs/1707.02069

[97] H. Zhang, L. Cheng, Y. Hao, and C.-w. Ngo, "Long-term leap attention, short-term periodic shift for video classification," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. ACM, Oct. 2022, p. 5773–5782. [Online]. Available: http://dx.doi.org/10.1145/3503161.3547908

[98] D. Shen, Y. Ji, P. Li, Y. Wang, and D. Lin, "Ranet: Region attention network for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[99] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[100] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3640–3649.

[101] K. Wang, X. Peng, J. Yang, D. Meng, and Y. Qiao, "Region attention networks for pose and occlusion robust facial expression recognition," *IEEE Transactions on Image Processing*, vol. 29, pp. 4057–4069, 2020.

[102] Y. Zhou, Z. Li, W. Yang, and P. Luo, "Hierarchical multi-scale attention for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2449–2461, 2019.

[103] H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, and L.-C. Chen, "Axial-deeplab: Stand-alone axial-attention for panoptic segmentation," 2020. [Online]. Available: https://arxiv.org/abs/2003.07853

[104] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 603–612.

[105] Z. Huang, X. Wang, Y. Wei, L. Huang, H. Shi, W. Liu, and T. S. Huang, "Ccnet: Criss-cross attention for semantic segmentation," 2020. [Online]. Available: https://arxiv.org/abs/1811.11721

[106] H. Wang, Y. Zhu, Z. Huang, X. Li, R. Luo, and D. Li, "Axial-deeplab: Stand-alone axial-attention for panoptic segmentation," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 108–126.

[107] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," 2018. [Online]. Available: https://arxiv.org/abs/1705.07750

[108] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages," *Information Fusion*, vol. 36, pp. 22–31, 2017.

[109] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 7464–7473.

[110] A. Miech, J.-B. Alayrac, P. Bojanowski, E. Grave, I. Laptev, and J. Sivic, "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 2630–2640.

[111] M. Chen, L. Li, W. Wang, R. Quan, and Y. Yang, "General and task-oriented video segmentation," 2024. [Online]. Available: https://arxiv.org/abs/2407.06540

[112] M. Li, S. Li, X. Zhang, and L. Zhang, "Univs: Unified and universal video segmentation with prompts as queries," 2024. [Online]. Available: https://arxiv.org/abs/2402.18115

[113] X. Li, H. Yuan, W. Li, H. Ding, S. Wu, W. Zhang, Y. Li, K. Chen, and C. C. Loy, "Omg-seg: Is one model good enough for all segmentation?" 2024. [Online]. Available: https://arxiv.org/abs/2401.10229

[114] J. You, A. Perkis, M. Gabbouj, and T. Ebrahimi, "Hierarchical semantic content analysis and its applications in multimedia summarization and browsing," 2011.

[115] Z. Liu, Y. Wang, and T. Chen, "Audio feature extraction and analysis for scene segmentation and classification," *Journal of VLSI Signal Processing*, vol. 20, 04 1998.

[116] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[117] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018. [Online]. Available: https://arxiv.org/abs/1804.02767

[118] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2625–2634.

[119] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014. [Online]. Available: https://arxiv.org/abs/1406.1078

[120] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," 2018. [Online]. Available: https://arxiv.org/abs/1711.07971

[121] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Tensor fusion network for multimodal sentiment analysis," *EMNLP*, 2017.

[122] R. Gao, R. Feris, and K. Grauman, "Listen to look: Action recognition by previewing audio," in *CVPR*, 2020.

[123] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," *ICML*, 2011.

[124] J. Arevalo, T. Solorio, M. Montes-y Gómez, and F. A. González, "Gated multimodal units for information fusion," *ICLR*, 2017.

[125] H. Akbari, L. Yuan, R. Qian, W.-H. Chuang, S.-F. Chang, B. Gong, and L. Luo, “Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text,” *NeurIPS*, 2021.

[126] B. Logan, “Mel frequency cepstral coefficients for music modeling,” in *International Symposium on Music Information Retrieval (ISMIR)*, 2000.

[127] L. Muda, M. Begam, and I. Elamvazuthi, “Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques,” *Journal of Computing*, vol. 2, no. 3, pp. 138–143, 2010.