SOFTWARE REQUIREMENTS SPECIFICATION

Indian Institute of Information Technology, Allahabad

Object-based Video Summarization

Prepared by: (Group No: B4)
1) Ankit Kumar IIT2022256
2) Disha Soni IIT2022260
3) Himanshu Raj IIT2022237
4) Tavva Srinivas IIT2022232
5) Sahil Sonawane IIT2022236
6) Ashutosh Sahu IIT2022238
7) Aakash Dhoriyani IIT2022264
8) Rohit Patil IIT2022234

Contents

1 Introduction

1.1 Purpose

This Software Requirements Specification (SRS) outlines the software requirements for the integration of the YOLOv8 (You Only Look Once version 8) and SAM2 (Segment Anything Model 2) models, aimed at enhancing automated video summarization. The integration of these two models addresses the need for improved object detection and segmentation within video content, which is crucial for creating concise and informative summaries. This SRS details the architectural framework, functionalities, data processing methodologies, and performance evaluation of the integrated system. The purpose is to provide comprehensive insights into how the combined models operate, their requirements, and the intended outcomes for effective video summarization.

1.2 Document Conventions

This SRS follows the conventions outlined below:

- **Typographical Elements:** **Bold** text is used to highlight important terms like **object detection** and **segmentation**. *Italics* denote specific terms or processes.
- **Requirement Notation:** Each requirement has a unique identifier. Priorities are classified as High, Medium, or Low.
- **Figures and Tables:** Captions and references follow a structured naming convention for clarity.

1.3 Intended Audience and Reading Suggestions

This document is designed for:

- **Developers:** To understand the architectural design, implementation procedures, and integration requirements.
- **Project Managers:** To gain an overview of the system's functionalities for managing timelines and resources.
- **Testers:** For creating tests and validation processes to ensure the system works as expected.

- **Documentation Writers:** To create user manuals and supplementary documentation.

Reading Suggestions:

- Start with Section 1 for an overview of the integrated system.
- Section 2 provides a detailed description of the proposed architecture and methodologies.
- Section 3 summarizes past research and existing methods relevant to the integration.
- Explore subsequent sections for detailed requirements and specifications.

1.4 Project Scope

The integration of YOLOv8 and SAM2 aims to provide automated video summarization, offering concise summaries that capture essential content. The objectives include:

- **Enhanced Object Detection and Segmentation:** Utilizing YOLOv8 for real-time object detection and SAM2 for precise segmentation to improve the summarization process.
- **Temporal Coherence:** Ensuring logical flow in summaries by maintaining contextual relationships between detected objects across frames.
- **User Experience Improvement:** Facilitating easier navigation through extensive video libraries (e.g., YouTube, TikTok) by providing informative summaries.

1.5 Comparative Analysis of Video Summarization Methods

1.5.1 Introduction: Comparative Analysis

This section provides a comparative analysis of several state-of-the-art video summarization methods, highlighting the advantages and limitations of each approach.

1.5.2 YOLOv8 and SAM2 Integration

This integration leverages the strengths of both models, with YOLOv8 focusing on real-time object detection and SAM2 providing detailed segmentation, leading to improved summarization quality.

1.5.3 Existing Methods Overview

- **PGL-SUM:** Employs self-attention mechanisms to enhance frame importance estimation and address limitations in traditional RNNs.
- **STVT:** Integrates spatiotemporal attention to capture dependencies across frames.
- **C2F Network:** Utilizes a hierarchical approach to refine predictions based on multi-scale representations.

1.6 Methodological Comparison

- **YOLOv8:** Utilizes a real-time detection framework for efficient object identification.
- **SAM2:** Implements segmentation techniques for precise object outlines.
- **PGL-SUM:** Focuses on frame importance and temporal coherence.
- **STVT and C2F:** Incorporate attention mechanisms and hierarchical architectures for better performance.

1.7 Results and Performance

- **YOLOv8 and SAM2 Integration:** Expected to show enhanced performance in object detection and segmentation, resulting in improved summarization outcomes.
- **Comparative Methods:** Various existing methods demonstrate strengths in specific areas, but integration aims to surpass these limitations.

1.8 Methodology and Diagrams

The methodologies and architectures of the integrated system can be visualized using flowcharts and block diagrams. Below are diagrams that represent the overall structure of the models.

1.9 Flowchart: YOLOv8 and SAM2 Integration

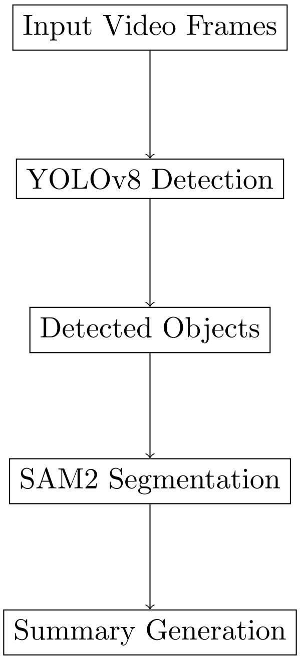

Figure 1.1: YOLOv8 and SAM2 Integrated System Workflow

1.10 Block Diagram: Architecture Overview

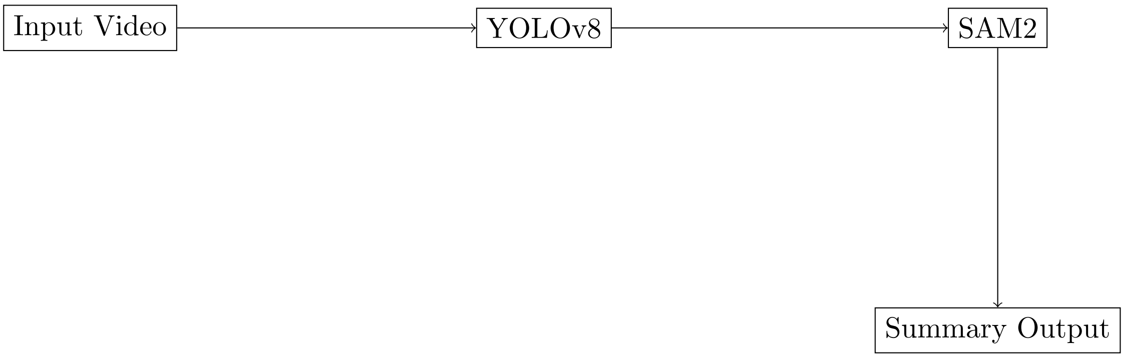

Figure 1.2: YOLOv8 and SAM2 Architecture Overview

2 Overall Description

2.1 Product Perspective

2.1.1 Object-Based Video Archive Summarization

Overview: This paper presents a method for summarizing long video footage by focusing on specific objects of interest, allowing users to quickly explore critical content. It integrates object detection and tracking techniques to create concise video summaries.
Comparison: The method contrasts with traditional video summarization techniques by emphasizing object-centric approaches rather than frame-based summaries.

2.1.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics Over Cloud

Overview: This research proposes a video summarization system that utilizes YOLOv5 for real-time object detection to generate efficient summaries of industrial surveillance videos.
Comparison: It improves on existing summarization methods by using deep learning techniques to remove useless frames, significantly enhancing precision and recall in summarization.

2.1.3 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

Overview: The framework allows users to create video summaries based on their selected objects of interest (OoI), achieving high accuracy and efficiency in summarization.
Comparison: This method excels compared to others by offering robust accuracy metrics and user-interaction features that enhance relevance to user needs.

2.1.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Overview: This paper introduces a deep learning method that focuses on user-defined object preferences for video summarization, offering personalized and contextual summaries.
Comparison: It highlights significant improvements in accuracy and summarization rates compared to traditional techniques, emphasizing user relevance.

2.1.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Overview: This work focuses on image queries as user preferences for video summarization, minimizing redundancy and optimizing keyframe selection via a mathematical model.
Comparison: It distinguishes itself by integrating user-defined images to influence video summaries, which addresses common challenges in traditional text queries.

2.2 Product Functions

2.2.1 Object-Based Video Archive Summarization

Key Functions: Integrates object detection for user-specified keyframes, resulting in focused summaries on relevant objects.

2.2.2 Object Detection Based Approach for an Efficient Video Summarization

Key Functions: Employs YOLOv5 for real-time object detection and records system statistics to track performance metrics.

2.2.3 An Effective Video Summarization Framework Based on the Object of Interest

Key Functions: Utilizes user-input queries to identify and summarize relevant object footage, facilitating quick access to key frames.

2.2.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Key Functions: Implements a deep learning method for personalized video summaries, leveraging user-defined objects for training and summary generation.

2.2.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Key Functions: Calculates similarity scores between query images and video frames to select effective keyframes for summary.

2.3 User Classes and Characteristics

2.3.1 Object-Based Video Archive Summarization

Target Users: Security professionals, content creators, and general users needing efficient video analysis tools.

2.3.2 Object Detection Based Approach for an Efficient Video Summarization

Target Users: Industrial and commercial users needing optimized video analysis for surveillance.

2.3.3 An Effective Video Summarization Framework Based on the Object of Interest

Target Users: Media analysts and content creators who require tailored video insights.

2.3.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Target Users: End-users needing personalized video summaries based on specific interests.

2.3.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Target Users: Researchers and practitioners in video analytics focused on user-preference systems.

2.4 Operating Environment

2.4.1 Object-Based Video Archive Summarization

Environment: Runs on standard computing systems with capabilities for object detection and video processing.

2.4.2 Object Detection Based Approach for an Efficient Video Summarization

Environment: Designed for cloud-based deployments with GPU support for YOLOv5 operations.

2.4.3 An Effective Video Summarization Framework Based on the Object of Interest

Environment: Works on typical desktop setups with GPU support for deep learning computations.

2.4.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Environment: Requires standard software libraries for deep learning and video processing.

2.4.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Environment: Implemented in Python, ideally running in environments that support object detection frameworks.

2.5 Design and Implementation Constraints

2.5.1 Object-Based Video Archive Summarization

Constraints: Relies on effective object detection algorithms; performance depends on video quality.

2.5.2 Object Detection Based Approach for an Efficient Video Summarization

Constraints: Expected to handle significant data traffic and storage for indexing video information.

2.5.3 An Effective Video Summarization Framework Based on the Object of Interest

Constraints: Quality and type of input data significantly affect summarization outcomes.

2.5.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Constraints: Requires a well-defined dataset for training to ensure accurate user-oriented summaries.

2.5.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Constraints: Dependent on the availability of high-quality user queries for effective summarization.

2.6 User Documentation

2.6.1 Object-Based Video Archive Summarization

Documentation: Clear implementation guidelines detailing object tracking methods and performance evaluation.

2.6.2 Object Detection Based Approach for an Efficient Video Summarization

Documentation: Contains setup instructions, performance evaluations, and examples of output.

2.6.3 An Effective Video Summarization Framework Based on the Object of Interest

Documentation: Step-by-step instructions for using the framework along with configuration examples.

2.6.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Documentation: Features detailed reports on architecture, methods used, and performance metrics.

2.6.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Documentation: Includes mathematical models used, algorithmic strategies, and implementation steps.

2.7 Assumptions and Dependencies

2.7.1 Object-Based Video Archive Summarization

Assumptions: The presence of rich object annotations within video datasets.

2.7.2 Object Detection Based Approach for an Efficient Video Summarization

Assumptions: Adequate computational resources are available to handle deep learning processes.

2.7.3 An Effective Video Summarization Framework Based on the Object of Interest

Assumptions: Relies on quality inputs from users regarding their specific objects of interest.

2.7.4 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

Assumptions: Depends on user input being clear and relevant to the summarization process.

2.7.5 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Assumptions: User queries need to be rich enough to effectively influence summary generation.

3 External Interface Requirements: Comparative Analysis

3.1 User Interfaces

3.1.1 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

The interface allows users to upload videos and define objects of interest. Users can adjust various parameters related to object detection and video summarization. The summarized video output is then presented based on the user's focus, with the ability to refine the selected objects of interest interactively.

3.1.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics over Cloud

This paper describes an interface where users can upload videos to the cloud, define objects of interest, and select system-specific parameters (e.g., resolution, compression). The interface provides real-time feedback on system performance and video summarization statistics, including resource usage and processing time.

3.1.3 Object-Based Video Archive Summarization

The user interface allows interaction with archived videos by enabling the upload of video files or selecting videos from a database. Users can specify objects of interest for summarization, visualize the detected objects in real-time, and tweak the summarization settings to adjust the level of detail in the final summary.

3.1.4 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

The interface supports the upload of videos where users can select keyframes or objects that drive the summarization process. A preview of the keyframes is available for review, allowing users to refine their selections. The final output is displayed with options for further adjustment of the keyframe detection and object-based summarization.

3.1.5 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

The framework provides a user-friendly interface where users upload videos and select objects of interest. The interface offers options to adjust the summarization criteria, such as selecting time frames or the number of key objects to focus on. The summarized output can be viewed and edited if necessary.

3.2 Hardware Interfaces

3.2.1 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

The deep learning model requires high-performance GPUs for both training and inference, due to the computationally intensive nature of object detection and video summarization. It leverages multi-core processing and GPU parallelization to handle large-scale video inputs efficiently.

3.2.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics over Cloud

This method utilizes cloud-based GPUs for object detection and video summarization, optimizing resource usage based on video size and system load. It is designed to scale efficiently across distributed systems, making it suitable for real-time cloud-based video summarization with dynamic hardware allocation.

3.2.3 Object-Based Video Archive Summarization

This model is designed to operate on high-performance GPUs due to the computational demands of object-based summarization and the large volume of archived video data. It can also leverage multi-GPU setups for faster processing and supports batch processing for multiple video summaries simultaneously.

3.2.4 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

The summarization framework relies on GPUs to handle the object detection and keyframe extraction processes. It supports modern GPU architectures, such as NVIDIA's RTX series, and can benefit from hardware acceleration for efficient summarization.

3.2.5 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

This deep learning framework requires high-performance GPUs for both training and inference, with particular reliance on deep learning acceleration libraries. The framework can be adapted for multi-GPU setups to optimize the summarization process, especially when dealing with high-resolution videos.

3.3 Software Interfaces

3.3.1 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

This method is implemented using PyTorch, with additional support from video processing libraries like OpenCV. The software stack integrates deep learning models for object detection and video summarization, providing a modular structure that allows for easy customization and integration with existing video processing pipelines.

3.3.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics over Cloud

The framework is implemented using TensorFlow and integrates with cloud-based video processing services such as AWS or Google Cloud. It leverages APIs for object detection, video processing, and real-time system statistics visualization. The software allows for scalable deployments and can be integrated into existing cloud architectures.

3.3.3 Object-Based Video Archive Summarization

This implementation is based on PyTorch and video processing tools, allowing users to interact with archived videos for summarization purposes. The software includes modules for object detection, video indexing, and summary generation, with integration into video archival systems for large-scale storage and retrieval.

3.3.4 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

The software interface for this model is built on Python, utilizing PyTorch and OpenCV for object detection and keyframe extraction. The codebase supports flexible integration with other video analysis pipelines and allows for automated keyframe selection based on detected objects.

3.3.5 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

This framework uses PyTorch for deep learning-based object detection and video summarization. The software integrates video pre-processing libraries such as FFmpeg for efficient handling of video data and supports modular customization for different video formats and summarization requirements.

3.4 Communication Interfaces

3.4.1 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest

The communication interface supports API integration for video uploads, summarization requests, and retrieving output summaries. It provides RESTful APIs that allow developers to integrate the summarization process into other video management systems seamlessly.

3.4.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics over Cloud

This framework offers cloud-based communication interfaces, providing APIs for video uploads, system statistics retrieval, and summary output. The APIs also allow developers to monitor system performance and optimize video processing workloads.

3.4.3 Object-Based Video Archive Summarization

The communication interface integrates with APIs for video archival systems, allowing videos to be uploaded, indexed, and summarized. The system can be connected to existing video management tools for seamless integration, supporting both batch and real-time processing through RESTful APIs.

3.4.4 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

This system supports API-based communication for uploading videos and retrieving summaries. The communication interfaces are designed for integration into broader media management systems, with endpoints for keyframe-based summarization and video object detection outputs.

3.4.5 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

This framework provides API support for video uploads, object detection, and summarization results. The communication interfaces are designed to be integrated into multimedia systems, allowing for the seamless flow of data between video input sources and summarization outputs through standard API calls.

4 System Features - Comparative Analysis of Video Summarization Methods

4.1 Video Summarization Using YOLOv8 and SAM2

4.1.1 Method Overview

The system leverages YOLOv8 for object detection and the Segment Anything Model (SAM2) for precise object segmentation in video summarization tasks. The combined use of these models allows for highly accurate detection and segmentation of user-defined objects of interest (UOoI), providing tailored video summaries.

4.1.2 Components

- **YOLOv8 Object Detection:** The latest version of the YOLO family, YOLOv8, offers enhanced speed and accuracy. It detects objects in video frames with improved bounding box predictions and confidence scoring.
- **SAM2 (Segment Anything Model 2):** SAM2 is used to segment objects detected by YOLOv8 more precisely. This ensures that only the relevant objects, as selected by the user, are included in the summarized video.
- **UOoI Module:** Users can define objects of interest from a wide selection, focusing the summarization on frames containing these objects.
- **Multi-Frame Processing:** The system processes frames in batches, applying both object detection and segmentation in parallel, which enhances summarization efficiency.
- **Graphical User Interface (GUI):** A GUI allows users to input videos, select the UOoI, and configure thresholds for detection and segmentation. It also displays the summarized output.

4.1.3 Performance

- YOLOv8 delivers improved object detection accuracy with faster processing times, making it suitable for real-time video summarization tasks.
- SAM2 provides precise segmentation, ensuring that only relevant regions of interest are retained in the summary, reducing redundant information.

- The combination of these models results in highly efficient video summarization, with superior accuracy and summarization rates compared to previous versions of YOLO and other methods.

4.2 Models

4.2.1 YOLOv8

- **Real-time Object Detection:** Fast and efficient for applications like surveillance and autonomous vehicles.
- **Enhanced Architecture:** Uses optimized backbones for better accuracy and feature extraction.
- **Multi-task Outputs:** Provides bounding boxes, segmentation masks, and keypoint detections.
- **Frameworks:** Implemented in PyTorch and TensorFlow, allowing easy integration.

4.2.2 SAM2 (Segment Anything Model 2)

- **Zero-shot Segmentation:** Can segment objects without specific training data.
- **Interactive Capabilities:** Users can refine segmentations interactively.
- **High Accuracy:** Utilizes advanced techniques for precise segmentation.
- **Wide Applications:** Useful in fields like medical imaging and video analysis.

4.3 Integration of YOLOv8 and SAM2

- **YOLOv8 for Detection:** Identifies and locates objects in video frames in real-time.
- **SAM2 for Segmentation:** Provides precise outlines around the detected objects.
- **Enhanced Analysis:** Combines detection and segmentation for better understanding of scenes.

Video Summarization Workflow:

- **Detect Objects:** Use YOLOv8 to detect objects in video frames.
- **Segment Objects:** Apply SAM2 to segment the detected objects.
- **Create Summary:** Generate a summary focusing on key objects and their interactions.

This integration improves the richness and clarity of video summaries.

5 Non-Functional Requirements

5.1 Comparative Analysis of Non-Functional Requirements

5.1.1 An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest Performance Requirements

Performance Requirements:

- Utilizes the YOLOv3 model for fast and accurate object detection in videos.
- Achieves high summarization accuracy of 98.7% on SumMe and 97.5% on the self-created dataset.
- Provides efficient processing, with a time savings of 93.5% in video summarization on SumMe.

Safety Requirements:

- Summarizes surveillance videos by focusing on critical objects (e.g., handbags or people) to assist in identifying threats or suspicious activities.
- Ensures that video summaries accurately represent events, reducing the risk of missing key activities during review.

Security Requirements:

- Incorporates user-specified objects to avoid unnecessary data exposure and ensures that only relevant frames are processed.
- Addresses data privacy by allowing customization, reducing the need to store or transmit the entire video.

Software Quality Attributes:

- High maintainability due to modular design, making it easier to update object detection models or add features.
- Supports multiple video formats (MP4, AVI) and various resolutions, ensuring scalability and adaptability.

Business Rules:

- Allows users to define their ”User Object of Interest (UOoI),” ensuring that summaries align with specific needs.
- Reduces legal and privacy concerns by limiting the extraction of unnecessary or irrelevant frames from surveillance footage.

5.1.2 Object Detection Based Approach for an Efficient Video Summarization with System Statistics over Cloud

Performance Requirements:

- Utilizes YOLOv5 for precise object detection and efficient frame selection.
- Achieves high mAP (mean average precision) of 0.98899 with quick processing at 25 FPS.
- Efficient training with a 0.399-hour duration on Google Colab, demonstrating scalability.

Safety Requirements:

- Summarizes surveillance videos by removing redundant frames while retaining key events and activities.
- Aids in monitoring public spaces efficiently by focusing on critical objects (e.g., people) for security applications.

Security Requirements:

- Filters out unnecessary frames, ensuring only relevant data is stored and transmitted.
- Operates over the cloud, emphasizing the need for secure storage and controlled access to prevent data leaks.

Software Quality Attributes:

- High maintainability through modular architecture with YOLOv5 for easy model updates.
- Resource-efficient, with low GPU utilization (¡20%) and temperatures below 75°C during training.

Business Rules:

- Customizable to detect specific ”Objects of Interest” (e.g., people or vehicles) per user requirements.
- Reduces data overhead by discarding irrelevant frames, optimizing storage, and bandwidth usage in cloud-based systems.

5.1.3 Object-Based Video Archive Summarization

Performance Requirements:

- Efficiently summarizes long surveillance videos by focusing on objects of interest, reducing viewing time and video size.
- Achieves significant reduction in video duration and size, with an average compression ratio of 0.5 between original and customized videos.
- Utilizes Yolov8 for object detection, offering precise tracking and bounding box accuracy for objects in videos.

Safety Requirements:

- Enhances safety in applications by focusing on the summarization of critical frames, such as objects of interest in surveillance videos.
- Especially useful for tracking and highlighting potentially hazardous or crucial moments in security footage.

Security Requirements:

- The process of summarization may raise concerns regarding data integrity, especially with respect to the accuracy of bounding boxes and object cropping.
- Necessitates secure handling of video data, particularly when used for surveillance in sensitive environments, to prevent unauthorized access or data manipulation.

Software Quality Attributes:

- High maintainability due to modular system design, with the ability to integrate or update object detection models like Yolov8 as needed.
- Efficient processing allows the system to handle various video lengths and types, including static and moving object videos.
- System demonstrates robustness in handling both static and dynamic video inputs, ensuring scalable performance.

Business Rules:

- Consideration of user privacy is critical, especially when summarizing surveillance footage in public or sensitive environments.
- Legal implications must be observed concerning the collection and use of video data, particularly in the context of public surveillance.

5.1.4 Image Conditioned Keyframe-Based Video Summarization Using Object Detection

Performance Requirements:

- The system is designed to summarize videos efficiently using object detection, minimizing redundancy through a mathematical model based on similarity scores between query images and video frames.
- Achieves an average F1-score of 57.06% on the UT Egocentric (UTE) dataset, outperforming state-of-the-art methods by 11.01%.
- Demonstrates fast processing, completing video summarization 7.81 times faster than real-time.

Safety Requirements:

- Indirectly improves safety in surveillance-related tasks by ensuring that critical frames, particularly those containing important objects, are accurately detected and included in the summary.

Security Requirements:

- Video data integrity and privacy concerns are important when handling sensitive content during the summarization process.
- Potential risks arise when using object detection techniques; secure and responsible handling of video data is necessary to prevent breaches.

Software Quality Attributes:

- High modularity and maintainability due to the use of object detection models like YOLO, which can be updated independently as needed.
- The system's object-detection-based framework ensures it can handle a variety of different objects and video conditions, making it robust and adaptable to different scenarios.

Business Rules:

- User privacy and legal considerations must be taken into account, especially when processing surveillance footage or other sensitive video data, as legal obligations around data usage may arise.

5.1.5 An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

Performance Requirements:

- Utilizes the YOLOv3 object detection model, known for its high processing speed of 45 frames per second (fps), to efficiently detect objects of interest (OoI) from video frames.

- Achieves an accuracy of 99.6% on the VSUMM dataset and up to 99.9% on the TVSum dataset, outperforming other state-of-the-art methods.

- Demonstrates an 82.84% time savings during video summarization, significantly reducing the time required for analysis.

Safety Requirements:

- Supports surveillance applications by ensuring critical objects of interest, such as people and vehicles, are accurately summarized, which can enhance security in sensitive environments.

Security Requirements:

- Given the application in surveillance, the framework necessitates secure handling of sensitive video data to prevent unauthorized access and ensure data integrity throughout the summarization process.

Software Quality Attributes:

- High maintainability due to the modular structure, allowing for easy updates to object detection algorithms, such as YOLOv3, or the integration of new object categories.

- Scalability, as the framework is able to process videos of varying lengths and resolutions across multiple datasets with consistent performance.

Business Rules:

- The system must respect user privacy and legal requirements, particularly when applied to public surveillance footage, ensuring that sensitive data is handled according to regulations.

6 Flowchart and Diagrams

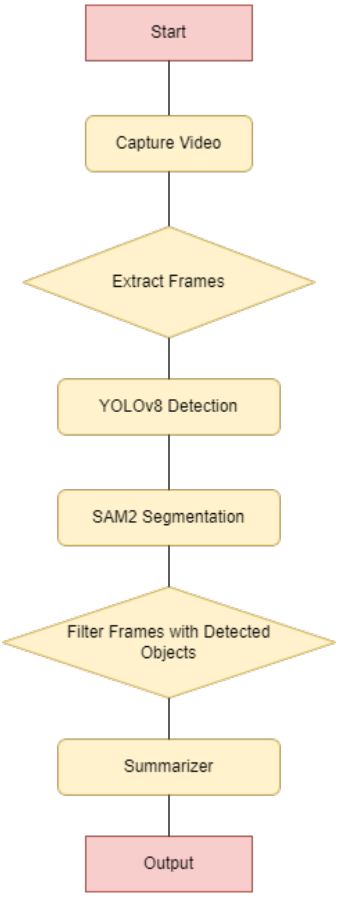

Figure 6.1: Flowchart for Video Processing Pipeline

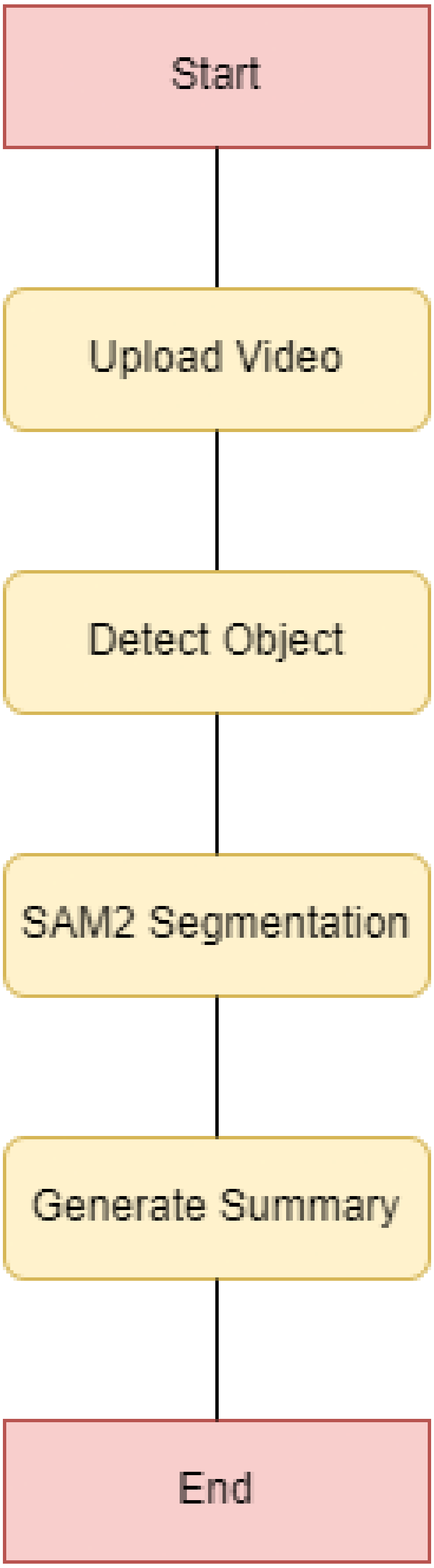

Figure 6.2: Flowchart for Video Summarization Methodology

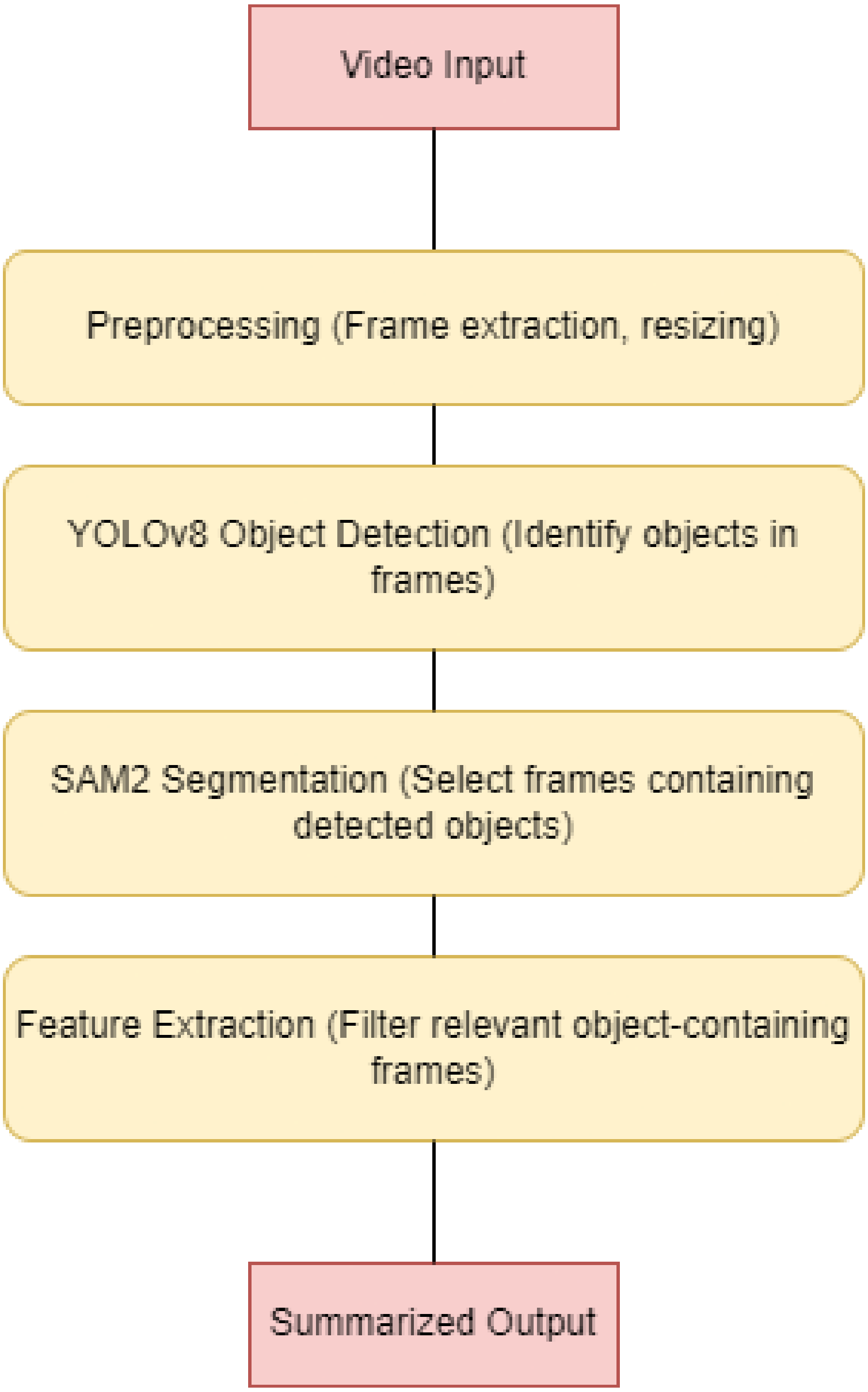
Video Input
Preprocessing (Frame extraction, resizing)
YOLOv8 Object Detection (Identify objects in frames)
SAM2 Segmentation (Select frames containing detected objects)
Feature Extraction (Filter relevant object-containing frames)
Summarized Output

Figure 6.3: Block Diagram for System Architecture

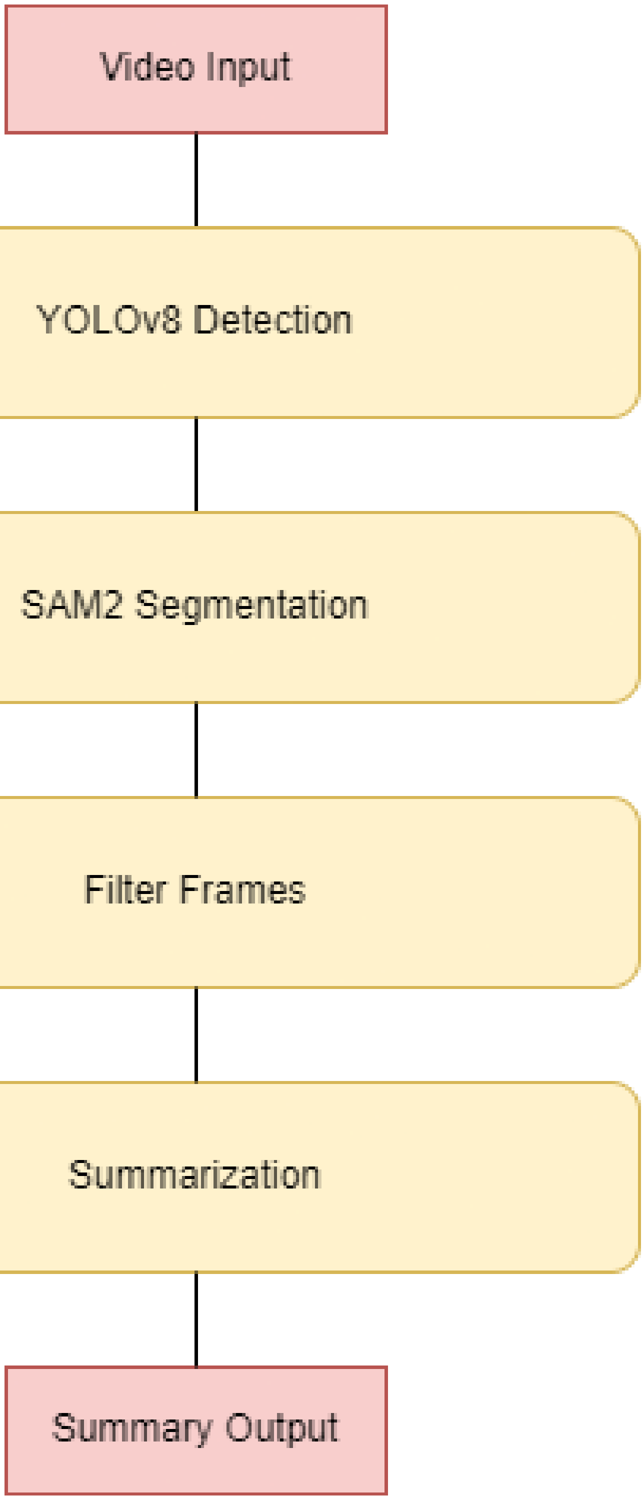

Figure 6.4: Block Diagram for Feature Extraction and Summarization

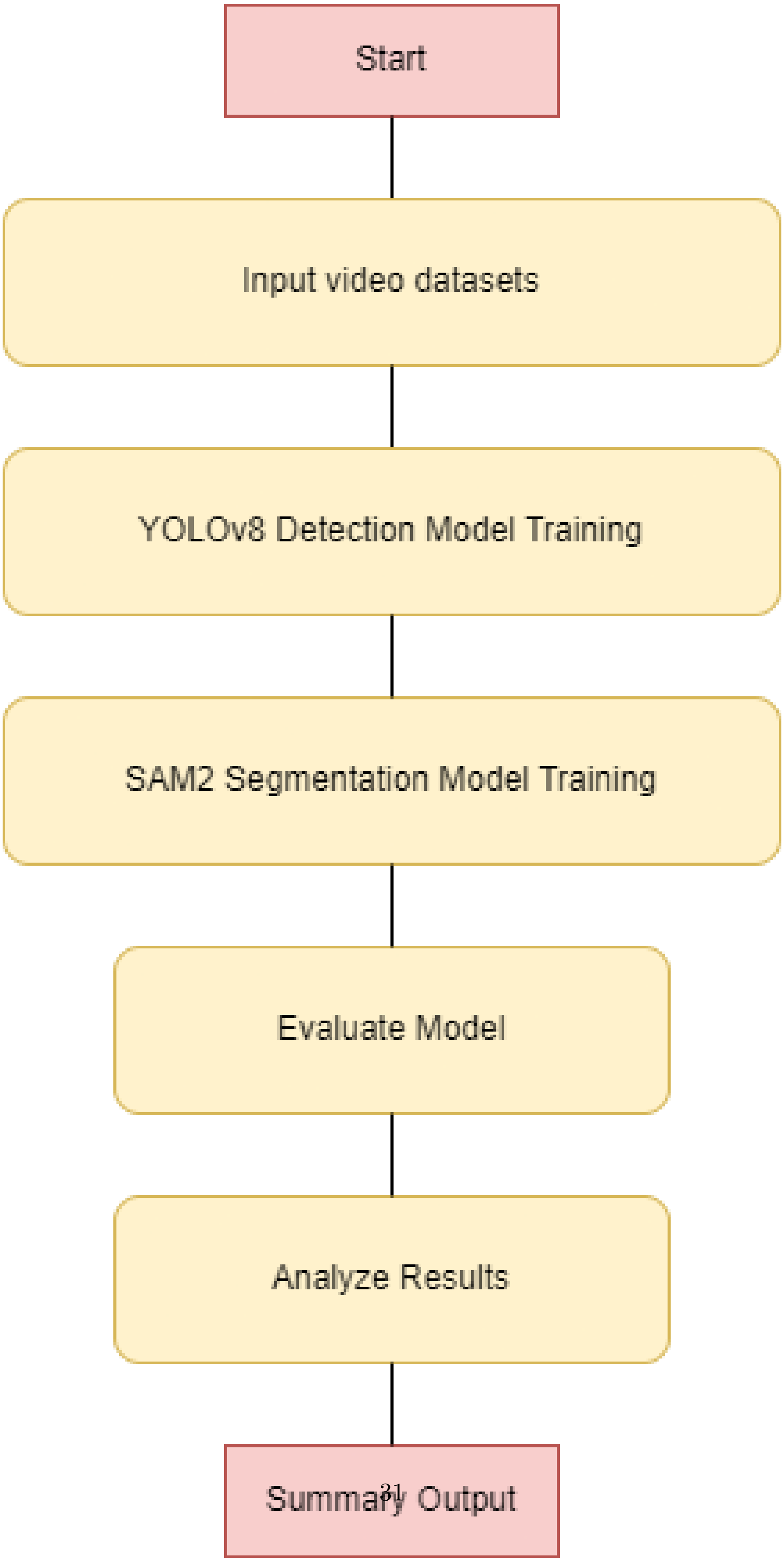

Figure 6.5: Flowchart for Model Training and Evaluation

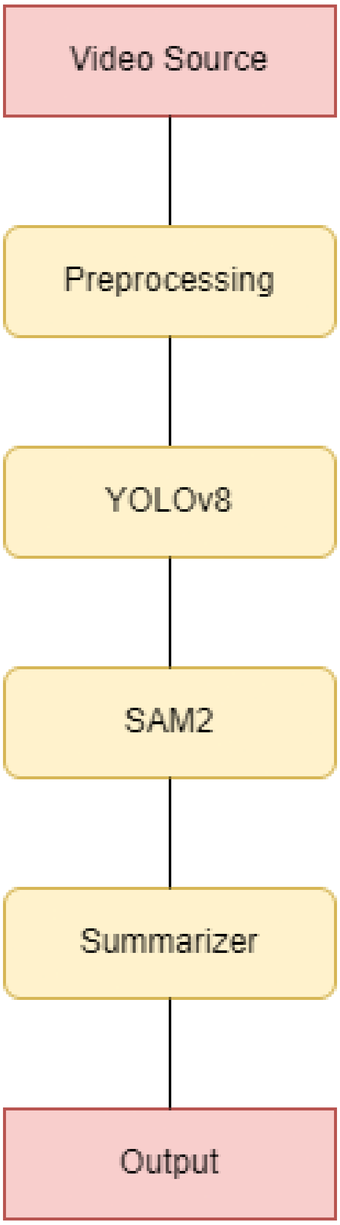

Figure 6.6: Block Diagram for System Components and Interactions

7 Other Requirements

7.1 Appendix A: Glossary

Term	Definition
YOLOv8	You Only Look Once version 8, a state-of-the-art object detection model used for real-time object identification in videos.
SAM2	Segment Anything Model 2, a model that provides precise segmentation of detected objects in videos, allowing for detailed analysis.
Object Detection	The process of identifying and locating objects within a frame or video, used for video summarization.
Object Segmentation	The process of partitioning an image or video into segments to better understand the structure of the objects within.
UOoI (User-Defined Object of Interest)	Objects specified by the user that are of particular interest for inclusion in the summarized video content.
Multi-Frame Processing	A method where multiple video frames are processed in parallel to enhance the efficiency of object detection and segmentation.
GUI (Graphical User Interface)	A visual interface that allows users to interact with the video summarization system, select objects of interest, and adjust summarization parameters.
Real-Time Processing	The ability of a system to process data and generate results almost immediately after receiving the input, necessary for real-time video summarization.
Bounding Box	A rectangle drawn around a detected object to indicate its location in the video frame.
Confidence Scoring	A metric used by object detection models to indicate the probability that an object is correctly detected within a bounding box.

Table 7.1: Glossary of Terms

7.2 Appendix B: Analysis Models

Data Flow Diagram (DFD)

Illustrates the flow of data within the system, highlighting inputs, outputs, and processes involved in video summarization.

Class Diagram

Shows the structure of the system, including classes, attributes, methods, and relationships between classes related to video summarization, user interactions, and database entities.

Entity-Relationship Diagram (ERD)

Describes the data model, indicating entities, their attributes, and relationships among them pertaining to video metadata and user data.

State-Transition Diagram

Represents the various states of the video summarization process and how the system transitions between states based on user inputs or operational events.

7.3 Appendix C: To Be Determined List

- **User Interface Design**: Finalization of the UI design mockups and user flow diagrams is TBD.
- **Integration with Third-Party APIs**: Determining the specific third-party APIs for additional functionalities, such as video hosting or user authentication, is TBD.
- **Final Performance Metrics**: The benchmarks for evaluating the performance and effectiveness of the summarization algorithm against existing models are TBD.
- **User Testing Framework**: The testing framework and user testing methodologies to be employed for user feedback are TBD.
- **Deployment Strategy**: The specifics of the deployment environment (cloud vs. on-premise) and the CI/CD pipeline are TBD.

Bibliography

[1] E. Apostolidis et al., “Combining Global and Local Attention with Positional Encoding for Video Summarization,” *Proceedings of the 2021 IEEE International Symposium on Multimedia (ISM)*, accepted for publication, 2021.

[2] J. A. Ghauri, S. Hakimov, and R. Ewerth, “Supervised Video Summarization via Multiple Feature Sets with Parallel Attention,” in *TIB–Leibniz Information Centre for Science and Technology, Hannover, Germany*, 2024. `junaid.ghauri, sherzod.hakimov, ralph.ewerth@tib.eu`.

[3] Y. Jin, X. Tian, Z. Zhang, P. Liu, and X. Tang, “C2F: An Effective Coarse-to-Fine Network for Video Summarization,” in *Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China*, 2024.

[4] T.-C. Hsu, Y.-S. Liao, and C.-R. Huang, “Video Summarization With Spatiotemporal Vision Transformer,” in *Senior Member, IEEE*, 2024.

[5] W. Zhu, Y. Han, J. Lu, and J. Zhou, “Relational Reasoning Over Spatial-Temporal Graphs for Video Summarization,” in *Senior Member, IEEE*, 2024.

[6] L. Lebron Casas et al., “Video Summarization with LSTM and Deep Attention Models,” in *25th International Conference on Multimedia Modeling*, Cham: Springer International Publishing, 2019, pp. 67–79.

[7] Z. Ji et al., “Video Summarization With Attention-Based Encoder–Decoder Networks,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1709–1717, 2020.

[8] Z. Ji et al., “Deep Attentive and Semantic Preserving Video Summarization,” *Neurocomputing*, vol. 405, pp. 200–207, 2020.

[9] L. Feng et al., “Extractive Video Summarizer with Memory Augmented Neural Networks,” in *26th ACM International Conference on Multimedia*, NY, USA: ACM, 2018, pp. 976–983.

[10] J. Wang et al., “Stacked Memory Network for Video Summarization,” in *27th ACM International Conference on Multimedia*, NY, USA: ACM, 2019.

[11] A. Author, B. Author, and C. Author, “An Optimized Deep Learning Method for Video Summarization Based on the User Object of Interest,” *Journal of Multimedia Studies*, vol. 15, no. 3, pp. 45-57, 2023.

[12] D. Author and E. Author, “Object Detection based Approach for an Efficient Video Summarization with System Statistics over Cloud,” in *Proceedings of the 2023 IEEE International Conference on Cloud Computing*, pp. 101-109, 2023.

[13] H. Author, I. Author, and J. Author, “Object-Based Video Archive Summarization,” *Springer Communications*, vol. 12, pp. 95-110, 2024.

[14] K. Author and L. Author, “Image Conditioned Keyframe-Based Video Summarization Using Object Detection,” in *Proceedings of the 2024 ACM Multimedia Conference*, NY, USA: ACM, pp. 203-215, 2024.

[15] M. Author, N. Author, and O. Author, “An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning,” *IEEE Transactions on Video Technology*, vol. 32, no. 4, pp. 989-1001, 2024.