Discover Artificial Intelligence

RESEARCH Open Access

Leveraging 5G networks for event video summarization via multimodal analysis

Alexandros Vrochidis[1,2*], Savvas Panagiotidis[3], Stathis Parcharidis[3], Nikolaos Dimitriou[1], Panagiotis Papaioannou[4], Evangelia Mililidaki[5], George Margetis[5], Dimitrios Tzovaras[1] and Stelios Krinidis[2]

*Correspondence:
Alexandros Vrochidis
avrochid@iti.gr
[1]Information Technologies Institute, Center for Research and Technology Hellas, 57001 Thessaloniki, Greece
[2]Department of Management Science and Technology, Democritus University of Thrace, 65404 Kavala, Greece
[3]Inventics - Hellas, 57001 Thessaloniki, Greece
[4]Department of Electrical and Computer Engineering, University of Patras, 26334 Patras, Greece
[5]Institute of Computer Science, Foundation for Research and Technology - Hellas, 71110 Heraklion, Greece

Abstract

With the rapid growth of video content across domains such as live streaming, remote education, surveillance, and virtual meetings, there is an increasing need for intelligent systems that can automatically analyze and summarize key events from video data. This paper presents an end-to-end framework that leverages 5G connectivity, Automatic Speech Recognition (ASR), Optical Character Recognition (OCR), and Large Language Models (LLMs) to generate concise, context-aware event summaries in the form of video highlights and textual overviews, eliminating the need to watch full videos. The system calibrates energy thresholds for ambient noise, extracts speech, and identifies keyframes for OCR, using HSV-based color histogram differencing. Redundant OCR outputs are filtered using a Levenshtein-based similarity threshold to retain unique content. TextRank is applied to reduce token load before invoking the OpenAI API for abstractive summarization, with per-minute summaries providing high-level insights. These summaries are overlaid by an Augmented Reality (AR) application, powered by a Network Application that receives streaming, performs instance segmentation to identify speakers and employs a decision-making module that determines the placement of the summaries within the user's field of view. Experimental evaluation on a private 5G testbed demonstrates smooth 4K streaming with low latency, stable Channel Quality Indicator (CQI), and no frame loss, confirming the system's scalability and effectiveness in dynamic network conditions.

Keywords Video event analysis, 5G Video streaming, Intelligent video systems, Video summarization, Multimodal content analysis, Text summarization

1 Introduction

With the rapid growth of video content across platforms ranging from surveillance systems and remote education to live streaming and online meetings, there is an increasing demand for intelligent systems that can automatically extract, interpret, and summarize meaningful events from video. In response to this need, this paper presents a novel methodology for event-based video analysis and summarization that enables users to quickly identify key highlights and determine the relevance of a video without watching it in full. Traditional video analysis often emphasizes visual content alone, overlooking valuable embedded information such as spoken dialogue and on-screen text.

By leveraging ASR and OCR, this approach captures rich multimodal data, enabling a deeper semantic understanding of video content. Recent advancements in LLMs further enhance this process by providing powerful tools for summarizing and contextualizing extracted information in a human-readable form. Additionally, the emergence of 5G networks facilitates real-time video streaming with minimal latency, creating new opportunities for deploying such intelligent summarization pipelines in dynamic, bandwidth-intensive environments. The proposed integrated system combines ASR, OCR, LLM-based summarization, and AR within a 5G-enabled framework to deliver accessible, scalable, and user-friendly video event summaries utilizing a Network Application developed as part of the FIDAL project [1].

Artificial Intelligence (AI) has become a cornerstone of modern video analysis, enabling systems to move beyond basic frame-by-frame inspection to a deeper, contextual understanding of visual content. AI techniques, particularly those involving machine learning and deep learning, allow for the automatic detection of objects [2], actions, scenes, and even emotions [3] within video streams. This level of analysis is critical in applications such as surveillance, content moderation, autonomous systems, and media summarization, where manual review is either impractical or inefficient. Furthermore, AI models can integrate multiple data modalities such as audio, text, and visuals to extract high-level semantic insights [4], significantly enhancing the accuracy and relevance of video interpretation. As video data continues to grow exponentially, AI-driven processing [5] becomes not just beneficial but essential for scalable, real-time, and intelligent video content understanding [6].

Despite significant advancements, several challenges persist in the application of OCR, ASR, and AR annotations within video analysis systems. OCR performance can be hindered by low-resolution frames, motion blur, diverse fonts, and complex backgrounds, making accurate text extraction difficult in real-world video streams. Similarly, speech recognition systems face challenges such as background noise, overlapping speakers, various accents, and domain-specific vocabulary, all of which can degrade transcription accuracy. AR annotations, while valuable for enhancing user interaction, must be precisely aligned with moving objects and scene dynamics in real-time, requiring robust tracking and spatial awareness under varying lighting, occlusion, and environmental conditions. Integrating these components seamlessly into a unified, real-time analysis pipeline remains a non-trivial task, especially when operating under the constraints of mobile or edge devices.

This work presents an integrated and automated video analysis system that leverages ASR and OCR to extract semantic content from video streams. By combining these modalities with advanced language models for summarization, the system enables users to access condensed, meaningful representations of video content without watching it in full. The solution not only generates textual summaries but also produces a short, trailer-style video capturing the highlights. In addition, it improves user interaction through AR by overlaying subtitles and key insights directly on the video content. The system further provides time-aligned summaries for each minute of video, offering users a granular understanding of the narrative progression. This multi-layered summarization approach improves video accessibility, saves valuable time, and supports more efficient content browsing and comprehension. The contributions of this work are:

- A multimodal automatic content extraction that combines ASR and OCR to extract both spoken and written information from videos for richer semantic understanding;
- A time-aligned summarization that provides per-minute summaries, enabling fine-grained insight into the video's timeline;
- An automated highlight generation that produces short, trailer-style video summaries that capture the most significant moments, aiding quick content appraisal;
- A Network Application designed for 5G-enabled, low-latency environments, running on the edge and composed by three key components: the Video Receiver, which receives the video stream; the video analyzer, which uses AI-based instance segmentation to identify speakers in real-time; and the decision maker, which selects the relevant subtitles or summaries and determines their spatial placement in the AR view;
- An AR interface that renders subtitles and key video information as overlays, enhancing the viewing experience of the user;
- Evaluation of speech recognition models on real-event videos in the Greek language, assessing their effectiveness in handling diverse, multilingual, and acoustically challenging content;

The remainder of this paper is structured as follows. Section 2 reviews the relevant literature. Section 3 outlines the system architecture for event video summarization, while Sect. 4 details the video analysis pipeline. Section 5 provides the results and performance evaluation. Finally, Sect. 6 concludes the paper with a summary of key findings, current limitations, and directions for future work.

2 Related work

In [7], a real-time multilingual speech recognition and speaker diarization system was developed using OpenAI's Whisper model, optimized for complex, multispeaker environments such as Taiwanese talk shows. The system achieved a low overall word diarization error rate of 6.96% demonstrating its effectiveness in handling frequent speaker changes and accented Mandarin speech. The Whisper model was also applied in [8] to develop an ASR system for individuals with dysarthria. Using the UA-Speech Corpus and a Bi-LSTM classifier, the system attained a word recognition accuracy of 59.78% highlighting its potential to enhance communication accessibility for speech-impaired users. In [9], a multi-encoder-decoder transformer was proposed to address the problem of switching between two or more languages in speech recognition, employing language-specific decoders to manage multilingual input effectively. Additionally, [10] introduced a memory-efficient transformer architecture that significantly reduced model parameters and improved training speed by over 50%, with inference accelerated by 1.35× compared to the baseline.

OCR is a key computer vision technology for extracting text from images [11] and videos, which is essential for tasks like license plate recognition. In [12], a YOLOv4-based system detected Bangladeshi vehicle plates, using Tesseract OCR for character recognition. It achieved 90.50% mean average precision and 14 fps on a TESLA T4 GPU, despite challenges like orientation, motion blur, and lighting conditions. OCR is also used for paper digitization. In [13], researchers converted electrocardiography into digitized signals and used OCR to remove characters and to communicate demographic information about patients. The used tool was developed with a predefined library of characters and

numbers. In [14], a Decoder-only Transformer model (DTrOCR) is proposed for text recognition, leveraging a pre-trained generative language model to simplify the traditional encoder-decoder architecture. Experimental results show that DTrOCR significantly outperforms state-of-the-art methods in recognizing printed, handwritten, and scene text in both English and Chinese. Appart from Tesseract, diffusion models [15] and encoders [16] have been used to reduce computational efficiency [17].

In [18], an extractive summarization framework that integrates Fuzzy Topic Modeling with BERT to enhance semantic and thematic representation is presented. Fuzzy logic enables more precise modeling of word-topic associations, improving the contextual relevance of extracted summaries. Except from BERT models, BART methods have been also used for text summarization. In [19], BRIO proposes a new training approach that models a non-deterministic distribution, allocating probability mass to various candidate summaries according to their quality. Graph-based methods are also used for text summarization. These methods include LexRank [20] and TextRank [21], which are foundational approaches that represent documents as graphs, where extractive summarization is framed as identifying the most central nodes. These methods are inspired by the PageRank algorithm [22], which ranks elements based on their importance within a network structure. Text summarization techniques have also been advanced by LLMs [23], such as OpenAI's GPT, which are capable of generating coherent and contextually relevant summaries through deep neural architectures. In [24], researchers proposed SumGPT, a model that combines T5 with a Vision Transformer to generate concise summaries of radiology reports. The model was evaluated against several baselines using metrics such as ROUGE-1, ROUGE-2, ROUGE-L, and BLEU to assess summary quality. In [25], researchers studied abstractive summarization for open-domain videos, focusing on generating fluent textual summaries from multiple source modalities, including video and audio transcripts. They evaluated their multi-source sequence-to-sequence models using metrics such as ROUGE, BLEU, and a proposed Content F1 to measure the semantic adequacy of the summaries.

AR has emerged as a powerful tool for enhancing user interaction through real-time information overlay on physical environments. Prior work has explored the integration of AR with context-aware systems to deliver dynamic, location-specific, and task-relevant content directly within the user's field of view. Studies have demonstrated its effectiveness across domains such as maintenance [26], education [27], and medical applications [28], where visual augmentation improves comprehension and task performance. Techniques such as marker-based tracking [29] and spatial mapping [30] have been employed to anchor virtual content accurately to real-world objects. Recent advances in AR hardware and computer vision algorithms have further enabled seamless and intuitive overlays, making AR a promising medium for information delivery in complex, data-rich settings.

Researchers have explored the synergy between 5G's high bandwidth and low-latency communication capabilities with edge computing's proximity to data sources, enabling real-time inference [31] and decision-making [32] at the network edge. Studies such as [33] have demonstrated how 5G-enabled edge infrastructures can support distributed AI frameworks, reducing the reliance on centralized cloud resources and mitigating the challenges of data transmission delays and network congestion. In [34], a novel approach to utilizing 5G edge node resources for both inferencing and training deep neural

network models in the context of massive Internet-of-Things services is proposed. In [35], a collaborative edge training is introduced. By avoiding centralized cloud training, the proposed framework enhances data privacy, reduces backhaul strain, and explores energy-aware scheduling and parallelism strategies to support scalable and eco-friendly edge-centric AI model training. These contributions highlight the growing role of 5G and edge computing as key enablers for practical, scalable AI systems in domains such as autonomous vehicles, smart manufacturing, and augmented reality. In [36], a Detect-to-Summarize framework is proposed that reformulates supervised video summarization as a temporal interest detection problem, addressing the limitations of frame-level regression approaches. The framework generates summarized clips and is evaluated on benchmark datasets such as SumMe and TVSum, demonstrating the advantages of incorporating temporal consistency into video summarization.

In [37], an AI-driven video processing approach is proposed to predict video popularity, which is measured by view counts using a linear regression model that leverages multimodal content analysis combined with early viewership metrics. In [38], researchers adopted a deep convolutional neural network for processing broadcast sports videos. To enhance multimodal analysis and video highlight generation, text preprocessing was applied to improve Tesseract's OCR accuracy, while Google Speech-to-Text API was used to transcribe spoken content. In [39], a work by the BBC explores how AI and machine learning can automate video production tasks, particularly for live events with limited crew availability. By using static UHD cameras and AI-driven visual analysis, the system can autonomously frame, sequence, and switch shots to simulate multicamera coverage, enabling efficient production of events.

In [40], an automated approach to generating soccer highlight clips by detecting logo transitions, scene boundaries, and optionally removing irrelevant scenes is proposed. In [41], a cloud-based video analytics framework designed for scalable and automated object detection and classification in recorded video streams is introduced. Challenges like the Ho Chi Minh City AI Challenge 2024 [43], highlight the growing demand for scalable, automated solutions in video analysis, emphasizing in processing of video data efficiently and accurately for tasks such as complex event retrieval. This work provides an integrated solution that automates complex video understanding tasks while enhancing the user experience in event-video platforms. The related works discussed in this paper are summarized in Table 1, which compares the proposed method with existing approaches across multiple key dimensions.

Table 1 Comparison of related work with the proposed approach

Study	PI	CGS	HD	TS	VS
[3] Audience analysis	✓				
[36] DSNet		✓	✓		✓
[37] Popularity prediction	✓	✓			
[38] Sports event summary			✓	✓	
[39] TV event analysis		✓	✓		
[40] Soccer event analysis			✓	✓	
[41] Video analysis in clouds		✓			
[42] Online video analysis	✓		✓		
Proposed video summarization	✓	✓	✓	✓	✓

*PI, platform integration; CGS, cross-genre support; HD, highlight detection; TS, text summary; VS, video summary (trailer)

3 System architecture

The System Architecture section presents the foundational components of the proposed system, with a focus on the integration of key AI modules alongside edge and cloud-based processing. It begins with an overview of the system architecture, outlining how various modules interact to support efficient data processing and intelligent decision-making. This is followed by a detailed breakdown of the edge processing and cloud integration strategy, highlighting how computational tasks are distributed to optimize performance and minimize latency. Finally, it explores the incorporation of AR for real-time annotation, where processed information is dynamically overlaid onto mobile devices or AR glasses, enhancing both the user experience and interaction with digital content in physical environments.

3.1 Design overview of the system architecture

The application workflow shown in Fig. 1 starts when a user uploads a video to the LiveMedia [42] platform (Video Block). Upon upload, the processing pipeline is initiated. The first step is speech recognition (ASR Block), which transcribes the audio content and aligns it with precise timestamps, enabling a detailed mapping of what is said and when. Following transcription, the TextRank [44] (Text Summary Block) analyzes the transcript to extract the most relevant sentences based on their significance and interconnectivity. These key sentences are then used to identify important video segments. Finally, these segments are compiled into a concise summary video, highlighting the most impactful moments.

In addition to automated summarization, the system offers user-driven customization (Adaptive Video Summarization Block). Users can define their available viewing time, and the summarization engine dynamically adjusts the video summary to fit within this constraint. If the initial summary exceeds the specified limit, the system intelligently refines the selection of key segments to create a more concise version while preserving the core message. This adaptive approach enhances time efficiency and accessibility, making it especially useful for scenarios where users need rapid insights from lengthy content, such as conference talks, interviews, or educational lectures.

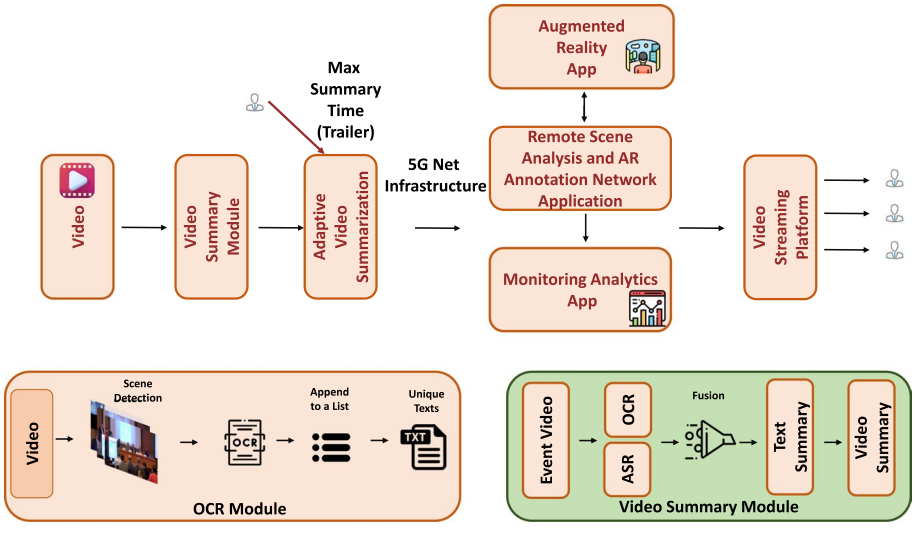

Fig. 1 Overview of the proposed methodology for event video summarization

Alongside speech recognition, the application employs the Tesseract OCR engine [45] to extract text directly from video frames (OCR Block). This is particularly useful in event recordings featuring slides or on-screen presentations, as the captured text often includes key points, titles, and supplementary information that enhances the spoken content. The transcribed speech is then refined using the TextRank algorithm [44], which filters out less relevant sentences to create a concise textual representation. This distilled content is used to generate a video summary via the OpenAI API [46], with an option to produce per-minute segment summaries for a more detailed overview. Additionally, OCR-derived text can be summarized independently, and due to its typically concise nature compared to speech transcripts, OCR-based summarization tends to be faster.

To enhance the quality of results, the system can integrate outputs from the ASR and OCR modules, producing a more comprehensive and informative summary. By fusing spoken content with visual text extracted from presentations, the system improves coverage and provides a fuller understanding of the video material. After processing audio and visual inputs, it generates a text-based summary along with a time-adaptive video summary aligned with the user's viewing availability. The summarized content and the video stream are first transmitted via 5G to the Remote Scene Analysis and AR Annotation Network Application (Remote Scene Analysis and AR Annotation Network Block), deployed at the edge, where the system analyzes the video in real-time and creates annotations linked to the active speaker. The combined information, including the annotation timing and placement, are then sent back to the AR Application (AR App Block), which overlays real-time contextual AR annotations, such as speech transcripts and per-minute summaries, directly onto the video. These overlays improve user comprehension and provide an engaging and comprehensible visual experience by emphasizing important details as they occur.

The Monitoring Analytics Service (Monitoring Analytics App Block) operates concurrently with the Remote Scene Analysis and AR Annotation Network Application, gathering performance and network data in real-time. It monitors key metrics such as latency, packet loss, processing time, and user interactions, providing valuable insights into both system efficiency and the impact of AR annotations. This information is essential for assessing the application's behavior under real-world conditions and verifying compliance with strict Key Performance Indicator (KPI) requirements for responsiveness and reliability. Once the video is enriched with AR overlays, it is uploaded to the LiveMedia [3] platform, where it becomes available for on-demand access by users.

3.2 Overview of the proposed AI modules

To enable ASR, the video is first converted into a WAV audio file. For the ASR, three different models are available for the user to choose, but more were tested to identify the most suitable option for a video platform that processes Greek and English audio. Google Speech Recognition [47] performed well in both languages without requiring language specification, which is ideal for automated systems. However, it lacks punctuation and requires an internet connection. PocketSphinx [48], developed by Carnegie Mellon University, is a lightweight, offline model that supports both languages, but it has lower accuracy and also omits punctuation. Vosk [49] operates offline and requires separate language models for Greek and English. It offers higher accuracy than PocketSphinx

and provides word-level timestamps, although it does not add punctuation. Whisper [50], available in tiny, base, and large versions, outperformed all other models in terms of accuracy, especially for Greek, and can process both short and long segments without pre-segmentation. It works offline and supports multilingual content effectively, making it the most suitable model for integration into the platform. If a model other than Whisper is used, timestamps are generated by first converting the video into a WAV file and then dividing it into segments of n seconds. Each segment is transcribed individually, and its corresponding timestamp is provided along with the transcription. The Word Error Rate (WER) was used to evaluate each model, with Whisper achieving the best overall performance. Its formula is given by:

$$\mathrm{WER} = \frac{S + D + I}{N} \times 100\%, \tag{1}$$

where S is the number of substitutions, D stands for the number of deletions, I is for the number of insertions, and N is for the total number of words in the reference transcription.

For the text recognition task, Tesseract [51] was chosen as the primary OCR tool due to its robustness, ease of use, and support for multiple languages. Tesseract is an open-source OCR engine that has been widely adopted in various applications because of its accuracy in recognizing printed and handwritten text across different fonts and languages. It was selected for its efficiency in processing text from images and its ability to integrate easily into the workflow. Moreover, Tesseract is known for its strong community support and continuous updates, which help improve its performance and keep it aligned with the latest advancements in OCR technology. These factors made Tesseract an ideal choice for the system's requirements, ensuring reliable and scalable performance in real-world applications.

For text summarization, the most important sentences are identified using the TextRank [44] algorithm, which supports both Greek and English. Based on the top-ranked moments determined by TextRank, video highlights can be generated using their corresponding timestamps and compiled into a highlights video using Moviepy [52]. Additionally, a full video summary can be produced with OpenAI's ChatGPT API [53]. Applying TextRank before using the API helps reduce token usage while maintaining summary accuracy. Furthermore, the system can optionally generate minute-by-minute text summaries for users seeking more granular insights. Text summarization significantly enhances user experience by providing quick insights into video content without requiring full viewing. It saves valuable time by highlighting key moments or summarizing each minute, allowing users to efficiently grasp the main ideas. This not only streamlines content consumption but also supports better decision-making about what to watch in depth.

For the summarization's evaluation, the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) and Bilingual Evaluation Understudy (BLEU) metrics were employed. These metrics require a ground truth for comparison. A subset of the test videos was analyzed to create a ground-truth dataset of human-generated summaries, against which the system's output was compared. The ROUGE metric, specifically ROUGE-1 and ROUGE-L, was used and is defined as follows:

$$\text{ROUGE} = \frac{N_{OU}}{T_U}, \tag{2}$$

where N_{OU} denotes the number of overlapping units (e.g., *n*-grams, words, or sequences) between the system-generated summary and the reference summary, T_U denotes the total number of units in the reference summary.

BLEU evaluates the precision of n-grams in the system-generated summary relative to one or more reference summaries, while applying a brevity penalty to discourage overly short summaries. BLEU is calculated as:

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{3}$$

where P_n is the precision of *n*-grams of order *n*, W_n is the weight assigned to n-grams of order *n* (typically uniform, e.g., $W_n = \frac{1}{N}$), and *BP* is the brevity penalty defined as:

$$BP = \begin{cases} 1, & \text{if} \quad c > r, \\ e^{1-\frac{r}{c}}, & \text{if} \quad c \leq r, \end{cases}$$

with *c* being the length of the candidate (system-generated) summary and r the effective reference summary length. By combining ROUGE and BLEU metrics, the evaluation captures both content coverage and *n*-gram precision, providing a robust assessment of summarization quality.

3.3 Edge processing and cloud integration

The proposed video summarization and AR annotation methodology supports flexible deployment models, allowing for edge and cloud-based computation. When deployed locally, the system can run on a PC or embedded computing device, processing video data and generating speech recognition transcripts, summaries, and metadata directly on-site. Once the content is prepared, it can be streamed in real-time over high-speed networks such as 5G, Wi-Fi, or other network infrastructures. This approach minimizes reliance on external connectivity during processing, offering faster response times and improved privacy for sensitive or real-time applications.

Alternatively, the entire pipeline can be deployed in the cloud, enabling scalability, centralized management, and easy access to powerful AI resources. Currently, the system is deployed on the Google Cloud Platform [42], where all video analysis is performed, including speech recognition, summarization, and metadata extraction. For AR-based use cases, the generated JSON files, containing subtitle and summary data-along with video frames, are then streamed to the Remote Scene Analysis and AR annotation Network Application, which is deployed at the edge. It receives the stream and associated subtitle and summary data, performs real-time scene analysis to identify active speakers, and employs a decision-making module to determine the spatial placement of the annotations. The resulting AR annotations are then streamed to the AR device over a 5G or similar high-speed connection for real-time rendering. This cloud-first model ensures robust performance, supports heavy workloads, and simplifies updates, making it ideal for large-scale or distributed deployments.

To support low-latency performance, especially in real-time applications like augmented reality, the system incorporates several optimizations. In the low-latency,

TCP-based setup, video frames are encoded in JPEG format and transmitted alongside lightweight JSON metadata, significantly reducing the data payload compared to raw video. This compressed structure facilitates faster delivery over networks such as 5G, minimizing lag in AR overlay rendering. The architecture also supports local processing at the edge, enabling the system to preprocess content on-site and transmit only essential metadata, further reducing network load and latency.

Additionally, the modular design allows selection between TCP for real-time data transfer or RabbitMQ for buffered, asynchronous delivery, depending on latency sensitivity. Frame resizing before transmission and the use of efficient data structures ensure that only the necessary information is sent, optimizing rendering performance on AR devices, even those with limited processing power. These design choices collectively ensure smooth, near-instantaneous playback and interaction in mobile AR environments.

3.4 AR annotation and mobile device integration

If the user has access to an AR-enabled device, such as a smartphone or smart glasses, the experience is further enhanced through real-time overlays that provide additional contextual information directly on top of the video. These AR annotations include both transcribed speech and concise summaries of each video minute, allowing users to quickly understand key moments without interrupting the viewing flow. This immersive layer of interaction deepens user engagement and supports faster content comprehension. Both AR Network Application and AR application are optimized for a wide range of screen sizes and device resolutions, ensuring a consistent and responsive AR experience across different platforms. Annotations are rendered in real-time with minimal latency, and interactions remain smooth even on mid-range mobile hardware.

The Remote Scene Analysis and AR annotation Network Application is integrated with a monitoring and analytics platform that visualizes key performance metrics such as bandwidth usage and latency. This connection enables real-time evaluation of system efficiency and responsiveness under varying network conditions. During testing on a 5G network, the app operated smoothly without performance issues, confirming its capability to handle high-speed, low-latency environments effectively. This seamless integration between AR-based summarization and system monitoring not only ensures technical reliability but also supports adaptive deployment across different network infrastructures. By enabling personalized, context-aware video experiences, this approach marks a significant step toward more intelligent and efficient content consumption.

The pipeline begins once the text summarization process is complete. The video stream and the associated data, which includes speech transcripts and summary data in JSON format, are forwarded to the Network Application. This Network Application processes the incoming data in real-time, identifies active speakers, and prepares spatially aligned annotations. The output is then sent to the AR device, allowing users to view synchronized visual content enhanced with contextual overlays.

The video streaming component is designed for adaptability and efficiency, supporting both Transmission Control Protocol (TCP) and RabbitMQ communication protocols to accommodate varying network and deployment scenarios. Each frame of the video is processed sequentially and matched with metadata, including speech recognition results and generated summaries, using accurate timestamps. If needed, frames are resized and

prepared with their associated annotations before transmission. In TCP-based configurations, frames are transmitted either as raw data or encoded JPEG images, alongside structured metadata. In RabbitMQ-based configurations mode, frames are base64-encoded and sent in JSON format via a message queue, ensuring reliable, asynchronous delivery. This robust streaming architecture enables real-time content delivery to AR devices while maintaining synchronization between video playback and annotation data.

This modular architecture enables seamless integration with a variety of back-end systems and AR display platforms. In the low-latency, TCP-based setup, the use of JSON metadata alongside compressed video frames ensures bandwidth-efficient transmission, making it particularly well-suited for real-time applications such as augmented reality overlays. The RabbitMQ-based mode supports scalable and distributed deployments, where video processing and transmission tasks may be decoupled and handled across multiple systems. Overall, this flexible design supports real-time video streaming enriched with contextual metadata, ensuring synchronized playback of video content with corresponding speech and summary annotations, which is an essential feature for delivering an engaging and informative experience in mobile AR applications.

4 AI processing pipeline

This section provides a detailed overview of the AI processing pipeline used in the system. It outlines the key components and methodologies involved in analyzing video content, including OCR, ASR, and the role of 5G technologies in enhancing performance and scalability. Each subsection will delve into the specific techniques, models, and workflows applied at different stages of the pipeline.

4.1 Video summary methodology

The AI processing pipeline begins by accepting a video URL as input and downloading the video for local analysis. The first module in the pipeline performs speech recognition. To enable this, the video is converted into a WAV audio file containing only its sound, which is then passed to the speech recognition system. The methodology supports three recognition models, including Vosk, Whisper, and Google Speech-to-Text. Among these, Whisper was selected for the experimental evaluation due to its superior performance on the dataset. Whisper operates locally and automatically adds punctuation, enhancing transcript readability. While the current implementation supports English and Greek, the system can be easily extended to include additional languages.

A dynamic filter is applied to adjust the energy threshold based on audio input from the source, allowing the system to account for varying ambient noise levels. This calibration ensures that speech is accurately detected under different environmental conditions. Following this, the speech recognition module processes the WAV file extracted from the video and returns transcribed speech along with timestamped segments. For models other than Whisper, the WAV file is divided into n segments, which are processed individually. This segmentation enables the system to associate each transcribed portion with its corresponding time in the video. Once speech recognition is complete, the system outputs a txt file and a json file, each containing the full transcription and segmented speech data extracted from the video.

After the ASR stage, the OSR process begins, starting with keyframe extraction, which is a technique used to identify representative frames that capture the essential

content of each video shot. Rather than analyzing every single frame, this method selects a subset of frames that best represent the visual content of each video shot. This significantly reduces computational load and storage requirements, making the processing more efficient. The keyframe extraction algorithm operates by analyzing color information between consecutive frames. It computes the color histogram difference and uses a predefined threshold defined by the user to detect shot boundaries. When the histogram difference between two frames exceeds this threshold *t*, the algorithm identifies the start of a new shot. Then it selects a representative frame from within the detected scene interval to serve as a keyframe. Although the selection is non-deterministic, it is constrained to the scene boundaries, ensuring that the chosen frame is always representative of the corresponding shot. The difference is calculated using the HSV color space instead of RGB, as HSV more closely aligns with human color perception and enables more intuitive and effective analysis of visual changes. The formulas used for computing the HSV color changes are:

$$\Delta H = \frac{1}{N_p} \sum_{i=1}^{N} |H_t(i) - H_{t-1}(i)|, \tag{4}$$

$$\Delta S = \frac{1}{N_p} \sum_{i=1}^{N} |S_t(i) - S_{t-1}(i)|, \tag{5}$$

$$\Delta V = \frac{1}{N_p} \sum_{i=1}^{N} |V_t(i) - V_{t-1}(i)|, \tag{6}$$

$$\Delta_{\mathrm{HSV\_avg}} = \frac{\Delta H + \Delta S + \Delta V}{3}. \tag{7}$$

$$\text{If } \Delta_{\mathrm{HSV\_avg}} \geq \text{threshold, then a scene change is detected.} \tag{8}$$

Where H_t, S_t, V_t are the hue, saturation, and value channels of the current frame at time *t*. H_{t-1}, S_{t-1}, V_{t-1} are the corresponding channel values of the previous frame. N_p is for the total number of pixels per channel.

After extracting a representative frame for each scene, using a low threshold to ensure even subtle slide changes are captured and to avoid analyzing the same slide multiple times, the OCR module is activated. For this task, the Tesseract [51] OCR model was selected due to its robust support for multiple languages, including Greek and English, as well as its computational efficiency, making it suitable for large-scale video processing.

Once each frame is processed by the OCR engine, the recognized text is extracted and stored in a list. To eliminate duplicate or near-duplicate entries, such as repeated slides that may have bypassed the scene detection module, a similarity threshold *St* is applied. This threshold compares the textual content of consecutive OCR outputs using a Levenshtein string similarity metric. Only the texts that differ by more than the threshold *St* are retained.

This post-processing step is crucial to improving the quality and uniqueness of the extracted data. It ensures that redundant slide content is filtered out, thereby reducing noise in the processing stage of summarization. In cases where identical slides remain on screen for extended periods without triggering a scene change, this method effectively

prevents duplication and preserves only the most relevant textual content. The algorithm for this processing is shown in Algorithm 1.

Require: Input video file V, scene change threshold S_{thresh}, text similarity threshold T_{thresh}
Ensure: Summarized text is generated from the visual content in the video
Detect scene boundaries in V based on visual change (e.g., histogram difference)
Extract a key frame from each detected scene and store in $K = \{k_1, k_2, \ldots, k_n\}$
Apply OCR to each key frame k_i to extract text t_i, forming $T_{raw} = \{t_1, t_2, \ldots, t_n\}$
Initialize $T_{filtered} \leftarrow \emptyset$
for each $t_i \in T_{raw}$ **do**
    if t_i has low similarity ($< T_{thresh}$) with all texts in $T_{filtered}$ **then**
        Append t_i to $T_{filtered}$
    end if
end for
Concatenate $T_{filtered}$ into a single document T_{input}
Send T_{input} to the OpenAI API for summarization
Receive summarized text $T_{summary}$
return $T_{summary}$

Algorithm 1 OCR-based video summarization pipeline

For the summarization task, the TextRank algorithm [44] is used due to its language-agnostic design, making it suitable for both Greek and English, and its computational efficiency, which is critical for scalable processing. TextRank is a graph-based ranking algorithm that identifies the most important sentences in a document by building a graph of sentence similarities and applying an iterative scoring mechanism, similar to PageRank. Unlike neural models, TextRank is unsupervised, requires no training, and performs well even in resource-constrained environments, making it ideal for lightweight extractive summarization in multilingual settings.

TextRank is used for generating video highlights by leveraging the timestamps associated with the top-ranked text summaries. These timestamps are used to extract the corresponding video segments, effectively creating a condensed summary video, which is similar to a trailer, and showcases the most important or informative parts of the original content. The TextRank algorithm is also utilized to reduce the number of tokens passed to the final video summarization stage. By pre-selecting the most relevant sentences, it minimizes the input size for downstream models, leading to lower computational costs and more efficient use of resources, which is especially important when using token-based APIs.

There are three text summarization approaches in the system. The first and fastest method relies solely on OCR output, as it bypasses the computationally intensive speech recognition step. The second approach uses speech recognition transcripts for summarization. The third, and most comprehensive method, fuses both the speech from ASR and the text from OCR, providing a richer summary that incorporates information from both the spoken content and the presentation slides. This fusion approach is ideal for users who want a more complete understanding that includes audio and visual data.

Additionally, the system includes a per-minute summarization module that takes the speech transcript along with its timestamps as input. It segments the transcript into one-minute intervals and generates a concise summary for each segment using OpenAI's API. This approach enables a time-aware overview of the content, allowing users to quickly grasp the key points discussed in each minute of the video. In the case of the AR application, users can view the summarization as an overlay displayed within the

specific region identified by the AR module. This contextual overlay enhances the user experience by providing concise, relevant information directly within their field of view, seamlessly integrating the summarized content with the real-world environment. Such a dynamic presentation allows for quick comprehension without interrupting the user's interaction with the physical surroundings.

4.2 5G network for data streaming

To evaluate the performance of the proposed system under high-speed connectivity conditions, a 5G network [54] infrastructure was utilized. The tests were performed at the University of Patras (UoP) 5G facilities [55], which are supported by both UoP and p-Net. This environment allowed for real-time transmission of video frames and metadata between the summarization system, the edge Network Application, and the AR devices with minimal latency. The high bandwidth and low-latency characteristics of the 5G network were critical in supporting the seamless delivery of augmented reality annotations synchronized with the video playback.

The project utilized the 5G network configured, deployed, and managed by Patras5G, which is an academic isolated non-public 5G network infrastructure offering experimentation capabilities for various application vertical interested in evaluating and validating such solutions. Patras5G infrastructure supports 5G-capable devices, which can connect to the deployed networks and offer multiple such devices. For legacy devices that are not 5G capable, Customer Premises Equipment (CPE) can be used. This CPE ensures seamless and reliable data transmission across the network, delivering the high-speed, low latency performance required for demanding applications such as video streaming and augmented reality, and also supports edge-based deployment of the Remote Scene Analysis and AR Annotation Network Application, allowing real-time tasks like speaker instance segmentation and annotation placement, which helps reduce delays and improve responsiveness. This robust infrastructure was crucial for the experimental evaluation of the project, providing the necessary bandwidth and stability to support data-intensive operations under realistic conditions.

Figure 2 illustrates a schematic of the end-to-end infrastructure used in this study. The experimental setup involved two laptops acting as a client and a consumer, both connected to the Patras5G network. This connection allowed them to interface directly with the deployed video analysis system. The primary objective of this testing was to evaluate the system's performance over a 5G network, focusing on aspects such as video streaming quality, latency, and the efficiency of real-time processing. By replicating real-world usage scenarios with these devices, valuable insights were gained into the methodology's scalability and responsiveness under the high-speed, low-latency conditions characteristic of 5G connectivity.

As for the latency optimization strategies, these included streaming the video with efficient video encodings, which reduce the amount of data transmitted and speed up frame delivery. In addition, the system's modular design supports asynchronous data transmission, especially when using RabbitMQ, enabling smoother data flow and reducing buffering delays. Furthermore, the high bandwidth and low latency characteristics of the Patras5G network inherently support faster data transmission, enabling real-time video streaming and augmented reality interactions with minimal lag. In addition to them, there is an option to resize frames before transmission, which decreases the data

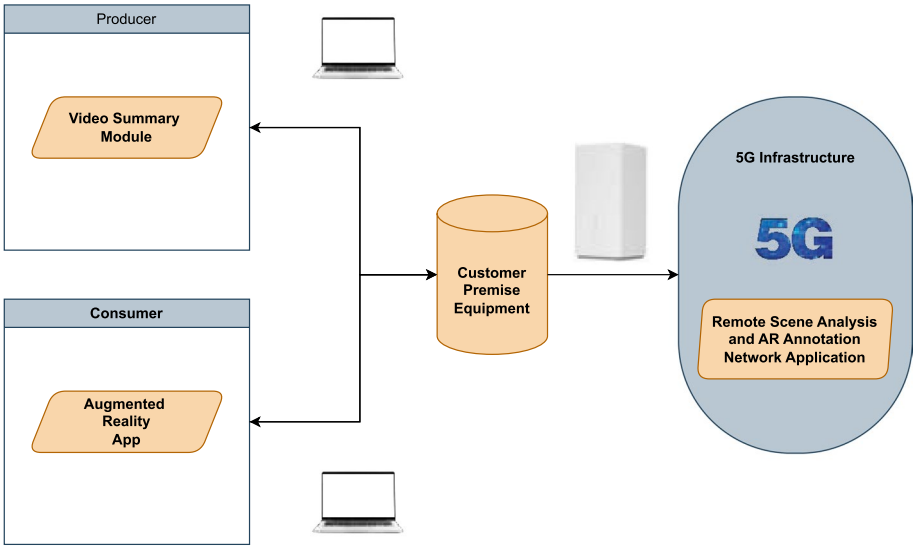

Fig. 2 Schematic figure about the end-to-end 5G infrastructure

volume. These combined approaches help ensure that latency remains low enough to maintain synchronization between video frames and corresponding text annotations, which is critical for a seamless user experience in augmented reality applications. Future iterations of the system could incorporate advanced adaptive streaming protocols or edge computing optimizations to further reduce latency and improve responsiveness under varying network conditions.

5 Experimental evaluation

This section presents the experimental evaluation of the proposed system, focusing on its performance, accuracy, and efficiency across key AI components such as OCR, ASR, text summarization, and AR integration. Various configurations and scenarios are tested to assess the system's effectiveness in real-world conditions, with special attention to language support, computational cost, and responsiveness across both edge and cloud environments. The results validate the practicality and robustness of the approach in video understanding and interactive applications.

5.1 Dataset

The dataset utilized in this study was sourced from the LiveMedia [3] platform, a real-time broadcasting service specializing in live event coverage. All videos were recordings of actual events captured and distributed by LiveMedia, ensuring realistic and diverse streaming conditions. For consistency in evaluating computational costs, the same typical conference event 10-minute HD video was used across all modules, serving as a common reference for comparing the performance of each modality. For all other evaluations, including ASR and OCR, a broader set of videos was employed, encompassing various event types, durations, and content complexity. This dataset included English, Greek, and bilingual videos, introducing linguistic diversity and reflecting real-world multilingual streaming scenarios. By leveraging authentic, heterogeneous video content, the evaluation captures the challenges of adaptive streaming across resolutions, languages, and visual complexity, enhancing the robustness and applicability of our

experimental findings. The videos used for the summarization evaluation were a subset of the test set, in both Greek and English, with a total duration of 116 min, sourced from various events.

5.2 Implementation details

The scene detection threshold was set at 8, carefully balancing sensitivity and accuracy to effectively identify scene changes while minimizing false positives. For the OCR results, a similarity threshold of 80% was applied using Levenshtein-based fuzzy matching to filter out duplicate or highly similar text entries, thereby enhancing the uniqueness and quality of the extracted textual data. The methodology was evaluated on a desktop system equipped with an Intel Core i5-10600K processor, 16 GB of RAM, and an NVIDIA GeForce GTX 1660 GPU.

For the network performance evaluation, two Apple MacBook devices were used, each featuring an Apple M2 chip and 8 GB of unified memory. This setup provided a consistent and controlled environment for assessing both computational efficiency and network behavior across various test scenarios. Notably, the application can run entirely on the CPU if no GPU is detected, ensuring compatibility with a wide range of hardware configurations. In addition, the Remote Scene Analysis and AR Annotation Network Application was deployed on two Ubuntu-based Virtual Machines (VMs) at the edge. One VM was GPU-enabled and dedicated to AI-based instance segmentation and video reception, while the second Virtual Machine handled annotation processing without requiring a GPU.

The 5G network used for all experiments and evaluations is the Patras5G, a private 5G testbed providing high-speed, low-latency connectivity. Its stable and controlled environment enabled reliable testing of video streaming performance, edge processing, and AR-based applications under realistic network conditions.

5.3 Video analysis results

Initial experiments in speech recognition were carried out using various methodologies to identify the most accurate and suitable model for integration into the platform. The models were tested on selected LiveMedia videos to assess their performance in real-world conditions. Evaluation was conducted in Greek and English, with a strong focus on Greek, given that most of the target video content is in this language.

Several parameters were considered when selecting the most suitable speech recognition model, including WER, punctuation support, and the ability to operate locally. Based on these criteria, three models were implemented in the final methodology: Vosk, Google Speech Recognition, and Whisper. Among them, Whisper demonstrated the highest accuracy, particularly in processing Greek audio, while also performing strongly with English, making it ideal for multilingual use. Its advanced capabilities allow it to process short and long video segments without needing pre-segmentation, streamlining the workflow. Moreover, its ability to function offline and deliver consistently accurate results makes Whisper a robust and reliable choice for integration into the platform. The results are presented in Table 2.

The speech recognition module is designed to leverage a GPU if available, but can also operate entirely on the CPU. For a typical 10-minute HD video, the most accurate `large_v3` model requires approximately 24 min to process, reflecting its higher

Table 2 Evaluation metrics for speech recognition models

Model	WER (%)	Punctuation	Offline Support
PocketSphinx	66.77	–	✓
Vosk	54.19	–	✓
Google speech recognition	34.40	–	–
Whisper	**5.09**	✓	✓

Bold values indicate the best result

computational demands. In contrast, the smaller model runs about three times faster, though with a trade-off in accuracy. During operation, the module utilized the full 6 GB GPU's VRAM, around 6,200 MB of RAM, and approximately 13.2% of CPU resources.

The OCR module performed well under the conditions presented by the event videos. Some challenges arose when both languages appeared simultaneously in a video. This issue was addressed by configuring the OCR to recognize both languages, which proved effective, though it slightly reduced overall accuracy. In cases where words were partially captured or incomplete, the summarization API still produced coherent summaries, as it was able to infer missing letters. This makes the approach especially well-suited for fast video summarization using OCR, eliminating the need for speech recognition and saving valuable time when speed is essential. For a typical 10-minute HD video, scene detection was completed in 31 s, achieving a speed of 450.31 frames per second. It used approximately 1,400 MB of RAM and 15.5% of the CPU.

In comparison, OCR processing took 2 min and 11 s to analyze 314 images, with an average speed of 2.17 frames per second. It required the same amount of RAM but utilized only 8% of the CPU. The results indicate that both modules are efficient, handling video frames with moderate CPU and memory usage. Their performance suggests they are well-suited for deployment on standard server infrastructure without requiring significant computational resources.

The summarization module, powered by the OpenAI API, processes a typical 10-minute HD video in approximately 4 s, with an estimated cost of 0.01 cents per run. For users seeking an offline alternative, deploying local large language models is a viable option, though it requires a powerful and efficient machine. The summarization results demonstrated strong accuracy across both languages tested, effectively capturing key video information and content, making the module a reliable tool for generating concise and informative summaries. The results of two different event videos are illustrated in Fig. 3. The module has been seamlessly integrated into the LiveMedia platform [3], making it readily accessible to its user base. This integration allows for ongoing, real-world evaluation and feedback directly from users, enabling continuous improvement based on practical usage and diverse application scenarios.

The AR application demonstrated strong performance during the experimental evaluation on the HD video, effectively identifying appropriate regions for text placement without issues. Frame rate analysis revealed a minimum of 21.7 fps and a maximum of 441.29 fps, with an average processing speed of 81.35 fps and a standard deviation of 52 fps. These results indicate that while there is some variability in performance depending on scene complexity, the application consistently maintained real-time or near-real-time responsiveness, confirming its suitability for interactive AR scenarios. The results of an annotated video are presented in Fig. 4.

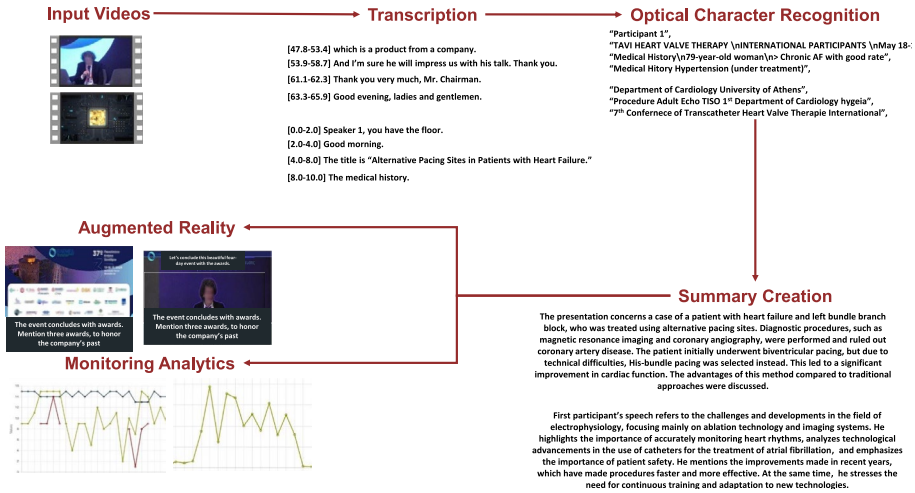

Fig. 3 Comparison of results produced from two separate event videos

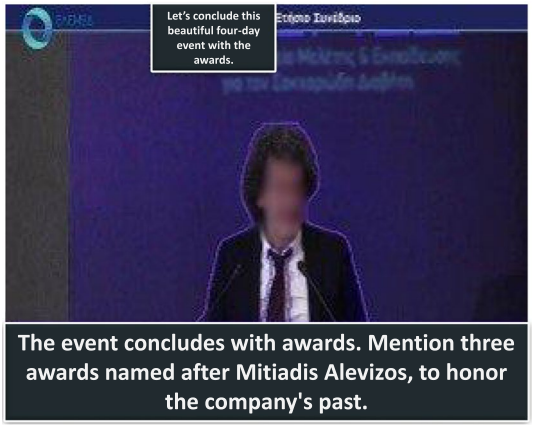

Fig. 4 Visualization of the AR annotations over a real video

The complete end-to-end processing of the module required approximately 29 min for a typical 10-minute HD video. This runtime encompasses all major components, including speech recognition, which is the heaviest task, scene detection, OCR, summarization, and AR integration. The result reflects a balanced trade-off between processing time and the quality and depth of the extracted information, making the module suitable for post-production analysis, archival, or content enrichment workflows.

Table 3 presents a comparative overview of the key AI technologies integrated into the system, highlighting their respective purposes, advantages, disadvantages, and suitability for different use cases. It includes various speech recognition models, such as Google Speech Recognition, Vosk, and Whisper. Each with distinct strengths, such as accuracy, offline capability, and multilingual support. Additionally, the table covers text summarization approaches, including the extractive TextRank algorithm and the abstractive OpenAI API, emphasizing their trade-offs between computational efficiency and summary quality. Finally, the Tesseract OCR module is featured for character recognition, valued for its open-source nature and multilingual support despite some sensitivity to image quality. This comprehensive comparison facilitates informed decisions on technology deployment based on operational requirements and resource constraints.

Table 3 Comparison of AI technologies used in the system

Technology	Purpose	Advantages	Disadvantages	Suitability
Google ASR	Speech recognition	High accuracy Auto language detection	Requires internet No punctuation support	Ideal for online environments with reliable connectivity
Vosk	Speech recognition	Offline capable Better accuracy than PocketSphinx	Separate models per language Slightly heavier than pocketsphinx	Good balance for limited connectivity scenarios
Whisper	Speech recognition	High accuracy Offline deployment Includes punctuation Multilingual	Higher resource requirements Slower than light-weight models	Best overall performer; ideal for final deployment
TextRank	Extractive text summarization	Unsupervised No training data needed Fast and scalable	Purely extractive May lack contextual coherence	Effective for quick, data-driven summaries
OpenAI API	Abstractive summarization	Context-aware Natural language generation High-quality summaries	Requires internet Cost per use Black-box model	Ideal for enhancing readability and coherence
Tesseract OCR	Character recognition	Open-source Multilingual support Customizable	Sensitive to image quality Requires preprocessing	Suitable for extracting slide and presentation text

For the evaluation of the summarization, using the texts produced after creating the ground truths, the average scores across all videos were 25.13% for ROUGE-1 F1, 2.73% for ROUGE-2 F1, 19.99% for ROUGE-L F1, and 1.54% for BLEU. Although these percentages appear low, the system-generated summaries often outperformed the human-generated ones in capturing key information. The relatively low scores are largely due to the use of different, yet more contextually appropriate, wording. These results indicate that the summarization approach is highly effective, accurately capturing the essential content of the videos. Furthermore, they demonstrate that incorporating speech as a source of information significantly enhances summary quality, producing coherent and contextually relevant outputs. Overall, the findings confirm that this method provides a reliable and robust way to generate meaningful summaries from spoken content.

5.4 5G streaming performance

To evaluate the methodology's performance over a 5G connection, a series of specific metrics were measured. All metrics were obtained from the same video source to maintain consistency and allow direct comparison across different performance indicators. The testing environment consisted of two laptops connected to the Patras5G network, with both the producer and consumer devices operating within the same network to ensure a controlled and reliable setup for accurate measurement and analysis. While multiple analyses were performed beyond 4K video streaming, this section focuses exclusively on the 4K streaming metrics for clarity and conciseness.

Table 4 presents the results, which indicate a generally robust performance for video streaming. The cell upload and download bitrates reflect strong network capacity, with the upload bitrate reaching 135.76 Mbps and the download bitrate peaking at 90.65 Mbps. The average CQI reported by the User Equipment (UE) to the gNodeB (gNb), indicating the quality of the connection between them from the UE perspective, had a

Table 4 Performance metrics for 5G video streaming evaluation

Metric	Maximum	Average	Standard Deviation
Cell upload bitrate	113.97 Mbps	89.67 Mbps	20.10 Mbps
Cell download bitrate	41.26 Mbps	12.88 Mbps	13.32 Mbps
CQI	15	11.22	5.78
Downloading bitrate	90.65 Mbps	3.28 Mbps	16.48 Mbps
PUSCH SNR	33.1 dB	21.04 dB	6.87 dB
Uploading bitrate	135.76 Mbps	35.66 Mbps	61.94 Mbps
Downloading MCS	27	17.18	6.69
Uploading MCS	27	6.89	3.40
User-centric upload bitrate	112.04 Mbps	89.66 Mbps	10.36 Mbps
App latency	212.58 ms	68.14 ms	11.40 ms
Frame loss	0	0	0
Application FPS	18.38	14.91	1.64

value of 11.22 and suggests overall good channel quality. From the user's perspective, an upload bitrate of 112.04 Mbps combined with zero frame loss highlights consistent performance and minimal packet loss throughout the test. The application latency, averaging 68.14 ms, further supports smooth and responsive video streaming well within acceptable limits. Modulation and Coding Schemes (MCS), defines the number of bits carried by every symbol during transmission and is an indicator of the signal quality. In general, high MCS means high quality and higher bitrates. The PUSCH Signal-to-Noise Ratio (SNR) refers to the uplink direction, representing the quality of the signal transmitted from the UE to the gNB. Additionally, the app's average FPS of 14.91 delivers a stable frame rate that ensures a solid 4K streaming experience. Although higher frame rates are typically preferred for even smoother playback, these results still indicate a satisfactory quality of viewing. Overall, the metrics confirm a strong foundation for video streaming, with latency and frame rates at levels suitable for effective live event streaming without noticeable disruptions.

An additional experiment was conducted to analyze KPIs using a different approach, by streaming the same video at three resolutions, including 4K, Full HD, and SD. Using the Remote Scene Analysis and AR annotation Network Application over the 5G PNET network, the setup ensured high-speed, low-latency conditions for reliable assessment. The analysis focused on network behavior, bitrate variation, and adaptive streaming performance. As shown in Fig. 5, each resolution was tested during separate time intervals. Higher resolutions naturally require greater bitrate due to the increased visual detail, confirming the link between video resolution and uplink bitrate in adaptive streaming scenarios.

In the next step, performance was evaluated across different video resolutions, revealing expected trends. Lower-resolution videos showed reduced latency due to smaller data loads, requiring less bandwidth and processing. Frame rates improved significantly at lower resolutions, as they demand fewer resources for decoding and rendering. Uplink MCS values remained relatively stable across all resolutions, indicating consistent transmission quality, with minor variations likely due to adaptive adjustments or network fluctuations. PUSCH SNR showed more variation at lower resolutions, likely caused by increased susceptibility to interference and noise due to reduced bandwidth usage. Lastly, CQI values slightly decreased and became more variable at lower resolutions, reflecting increased sensitivity to channel conditions and adaptive resource allocation. The results are shown in Table 5.

Fig. 5 Uplink bitrate variations during multiple resolution video streaming experiments

Table 5 Performance metrics across different video resolutions

Metric	4K	Full HD	SD
Application FPS			
Min	4.70	6.35	21.72
Max	18.38	58.06	441.29
Average	14.91	37.33	81.35
Std. Dev	1.64	9.11	52.00
Uplink MCS			
Min	1.7	4.5	4.5
Max	27	24	22
Average	6.89	6.93	6.82
Std. Dev	3.40	3.73	2.80
PUSCH SNR (dB)			
Min	5.8	3.4	2.6
Max	33.1	34.6	30.6
Average	21.04	20.75	20.51
Std. Dev	6.87	6.96	8.21
CQI			
Min	0	0	0
Max	15	15	15
Average	11.22	11.02	10.98
Std. Dev	5.78	6.77	6.47

6 Conclusion

In conclusion, a comprehensive video analysis, including speech recognition, text extraction, summarization, and AR enhancement can be effectively and efficiently achieved using the proposed methodology, making it a powerful solution for enriching and understanding video content at scale. The evaluation of each module demonstrated strong overall performance and accuracy, validating their suitability for integration into the LiveMedia platform. The speech recognition component, particularly the Whisper model, showed the highest accuracy in both Greek and English, with robust handling of long segments and offline capability, making it ideal for multilingual and resource-flexible environments. The OCR module performed reliably across various conditions, including bilingual content, with minor recognition issues that were effectively mitigated through configuration. The summarization module, delivered fast and coherent summaries with high linguistic accuracy. The scene detection module was notably fast

and efficient, requiring minimal computational resources, and the AR module consistently achieved real-time or near-real-time frame rates, demonstrating its effectiveness in dynamically placing text overlays.

Collectively, the modules achieved a balanced trade-off between processing time and output quality, supporting their use in both automated content analysis and enhancement workflows. The module's integration within the LiveMedia platform facilitates direct access for end-users, providing valuable real-world usage data and feedback. This continuous user-driven evaluation is essential for refining the module's performance and ensuring its effectiveness across a wide range of practical applications. In contrast, the smaller model runs about three times faster, though with a trade-off in accuracy. The summarization evaluation demonstrates that the proposed method effectively captures the key content of videos, even when using different, more contextually appropriate wording than human summaries. With a ROUGE-1 F1 of 25.13%, incorporating speech as a source of information significantly enhances the coherence and relevance of the generated summaries. Overall, the results confirm that this approach provides a reliable and robust method for producing meaningful summaries from spoken content.

The evaluation of video streaming performance across multiple resolutions over the 5G PNET network demonstrated that the proposed methodology effectively adapts to varying conditions while maintaining a high level of service quality. Key performance indicators such as latency, frame rate, and uplink bitrate consistently reflected the expected trade-offs associated with different video resolutions. Lower-resolution streams benefited from reduced latency and higher frame rates, while higher resolutions naturally demanded greater uplink bitrate due to increased visual complexity. Despite these variations, metrics such as uplink MCS and CQI remained stable, confirming the robustness of the network and the methodology's ability to support adaptive video streaming without significant degradation in quality. These results highlight the system's reliability for real-time streaming in diverse use cases, from high-definition broadcasting to low-bandwidth environments.

Additionally, the proposed methodology significantly enhances access to knowledge by empowering users to effortlessly discover videos that closely align with their interests and needs. This targeted approach saves valuable time and ensures that users engage with more relevant and meaningful content. By streamlining the search and discovery process, the methodology contributes to a more efficient and personalized learning experience, making it easier for users to access high-quality information in a rapidly growing digital landscape.

While the proposed system demonstrates strong potential, there are some limitations to consider. The accuracy of OCR and ASR components can be influenced by factors such as low video quality or background noise, which may occasionally impact the precision of extracted information. Additionally, although the combination of edge and cloud computing enhances scalability and responsiveness, variations in network connectivity and the capabilities of edge devices can sometimes affect real-time performance. These areas offer valuable opportunities for further refinement and optimization in future developments.

In future research, the impact of super-resolution techniques on improving OCR performance will be explored. By enhancing image resolution, super-resolution methods could potentially provide more detailed text features, leading to improved recognition

accuracy, particularly in cases where the input images are of lower quality or contain noise. In addition, with minor changes, the methodology could be applied to the automatic semantics' generation, further enhancing the user experience. This approach could significantly improve the system's ability to interpret context and meaning, leading to more intelligent, context-aware interactions and a deeper understanding of user inputs.

Acknowledgements
This project has received funding from HORIZON programme under grant agreement No 101135556 (project INDUX-R). This publication reflects only the authors' views. The European Commission is not responsible for any use that may be made of the information it contains.

Author contributions
Conceptualization, A.V. and N.D.; methodology, A.V. and N.D; software, A.V. and E.M.; validation, A.V., P.P, E.M, N.D, S.K.; formal analysis, N.D., P.P, S.P, S.P, D.T, S.K.; investigation, S.K., P.P, G.M, E.M; resources, N.D; data curation, N.D.; writing—original draft preparation, A.V.; writing—review and editing, A.V.; visualization, A.V.; supervision, N.D. and S.K.; project administration, N.D, S.P, S.P, D.T; funding acquisition, N.D. and S.P. All authors have read and agreed to the published version of the manuscript.

Funding
This research received no external funding.

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethical approval
This study did not involve human participants, animals, or sensitive data requiring ethical approval.

Consent to participate
Not applicable.

Consent to publish
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 4 July 2025 / Accepted: 25 November 2025
Published online: 12 December 2025

References

1. Papaioannou P, et al. Experiences and challenges on testing and validating network aware ppdr xr applications for 5G advanced networks. 2024 IEEE Future Networks World Forum (FNWF) 2024;204–209.
2. Wu Z, Gehrig M, Lyu Q, Liu X, Gilitschenski I. Leod: Label-efficient object detection for event cameras. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition 2024; 16933–16943.
3. Vrochidis A, et al. A multi-modal audience analysis system for predicting popularity of online videos. In: Proceedings of the 22nd engineering applications of neural networks conference. 2021; 465–476.
4. Guo J, Lu A, Wu Z, Wang Z, Liang C. Who, what and where: Composite-semantics instance search for story videos. In: IEEE transactions on image processing. 2025.
5. Thareja R, et al. Video analysis engine for predicting effectiveness. In: International conference on pattern recognition. 2025; 97–112.
6. Chen L, et al. Sharegpt4video: improving video understanding and generation with better captions. Adv Neural Inf Process Syst. 2024;37:19472–95.
7. Lyu KM, Lyu RY, Chang HT. Real-time multilingual speech recognition and speaker diarization system based on whisper segmentation. PeerJ Comput Sci. 2024;10:e1973.
8. Rathod S, Charola M, Patil HA. Transfer learning using whisper for dysarthric automatic speech recognition. In: International conference on speech and computer. 2023; 579–589.
9. Zhou X, Yılmaz E, Long Y, Li Y, Li H. Multi-encoder-decoder transformer for code-switching speech recognition. arXiv preprint arXiv:2006.10414 2020.
10. Winata GI, Cahyawijaya S, Lin Z, Liu Z, Fung P. Lightweight and efficient end-to-end speech recognition using low-rank transformer. In: ICASSP 2020 - 2020 IEEE international conference on acoustics, speech and signal processing (ICASSP). 2020; 6144–6148.
11. Anakpluek N, et al. Improved tesseract optical character recognition performance on thai document datasets. Big Data Res. 2025;39:100508.
12. Onim MSH, Akash MI, Haque M, Hafiz RI. Traffic surveillance using vehicle license plate detection and recognition in bangladesh. In: Proceedings of the 2020 11th international conference on electrical and computer engineering (ICECE). 2020; 121–124.

13. Memon J, Sami M, Khan RA, Uddin M. Handwritten optical character recognition (OCR): a comprehensive systematic literature review (SLR). IEEE Access. 2020;8:142642–68.
14. Fujitake M. Dtrocr: Decoder-only transformer for optical character recognition. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision (WACV). 2024; 8025–8035.
15. Fujitake M. Diffusionstr: Diffusion model for scene text recognition. In: 2023 IEEE international conference on image processing (ICIP). 2023; 1585–1589.
16. Atienza R. Vision transformer for fast and efficient scene text recognition. In: International conference on document analysis and recognition. 2021; 319–334
17. Du Y, et al. Svtr: Scene text recognition with a single visual model. arXiv preprint arXiv:2205.00159 2022.
18. Onan A, Alhumyani HA. Fuzzytp-bert: Enhancing extractive text summarization with fuzzy topic modeling and transformer networks. J King Saud Univ Comput Inf Sci. 2024;36:102080.
19. Liu Y, Liu P, Radev D, Neubig G. Brio: Bringing order to abstractive summarization. arXiv preprint arXiv:2203.16804 2022.
20. Deo S, Banik D. Text summarization using textrank and lexrank through latent semantic analysis. In: 2022 OITS international conference on information technology (OCIT). 2022; 113–118.
21. Liu W, et al. Automatic text summarization method based on improved textrank algorithm and k-means clustering. Knowl-Based Syst. 2024;287:111447.
22. Yang M, Wang H, Wei Z, Wang S, Wen JR. Efficient algorithms for personalized pagerank computation: a survey. In: IEEE transactions on knowledge and data engineering. 2024.
23. Shakil H, et al. Evaluating text summaries generated by large language models using openai's gpt. In: 2024 International conference on machine learning and applications (ICMLA). 2024; 951–956.
24. Sultan T, Rony MAT, Islam MS, Alshathri S, El-Shafai W. Sumgpt: a multimodal framework for radiology report summarization to improve clinical performance. IEEE Access. 2025;13:15929–45.
25. Palaskar S, Libovický J, Gella S, Metze F. Multimodal abstractive summarization for how2 videos. arXiv preprint arXiv:1906.07901 2019 .
26. Xu F, Nguyen T, Du J. Augmented reality for maintenance tasks with chatgpt for automated text-to-action. J Constr Eng Manag. 2024;150:04024015.
27. Koumpouros Y. Revealing the true potential and prospects of augmented reality in education. Smart Learn Environ. 2024;11:2.
28. Hu X, Cutolo F, Iqbal H, Henckel J, Baena FRY Artificial intelligence-driven framework for augmented reality markerless navigation in knee surgery. In: IEEE transactions on artificial intelligence. 2024.
29. Firdaus MB, et al. Penerapan metode marker based tracking augmented reality pesut mahakam. Jurnal Teknoinfo. 2022;16:20.
30. Balakrishnan S, Hameed MSS, Venkatesan K, Aswin G. Interaction of spatial computing in augmented reality. In: 2021 7th International conference on advanced computing and communication systems (ICACCS). 2021;1, 1900–1904.
31. Jere S, Wang Y, Aryendu I, Dayekh S, Liu L. Bayesian inference-assisted machine learning for near real-time jamming detection and classification in 5g new radio (nr). IEEE Trans Wireless Commun. 2024;23:7043–59.
32. Priyanka A, Gauthamarayathirumal P, Chandrasekar C. Machine learning algorithms in proactive decision making for handover management from 5g & beyond 5g. Egyptian Inf J. 2023;24:100389.
33. Li E, Zeng L, Zhou Z, Chen X. Edge ai: On-demand accelerating deep neural network inference via edge computing. IEEE Trans Wireless Commun. 2019;19:447–57.
34. Sung J, Han SJ. Use of edge resources for DNN model maintenance in 5g IoT networks. Clust Comput. 2024;27:5093–105.
35. Zeng L, Ye S, Chen X, Yang Y. Implementation of big AI models for wireless networks with collaborative edge computing. IEEE Wirel Commun. 2024;31:50–8.
36. Zhu W, Lu J, Li J, Zhou J. Dsnet: a flexible detect-to-summarize network for video summarization. IEEE Trans Image Process. 2021;30:948–62.
37. Vrochidis A, et al. Video popularity prediction through fusing early viewership with video content. Comput Vis Syst ICVS. 2021;2021:12899. https://doi.org/10.1007/978-3-030-87156-7_13.
38. Khan AA, Shao J, Ali W, Tumrani S. Content-aware summarization of broadcast sports videos: an audio–visual feature extraction approach. Neural Process Lett. 2020;52:1945–68.
39. Wright C, et al. Ai in production: Video analysis and machine learning for expanded live events coverage. In: Proceedings of the 2023 ACM international conference on interactive media experiences workshops. 2023; 77–78.
40. Valand JO, et al. Ai-based video clipping of soccer events. Mach Learn Knowl Extr. 2021;3:990–1008.
41. Anjum A, Abdullah T, Tariq MF, Baltaci Y, Antonopoulos N. Video stream analysis in clouds: an object detection and classification framework for high performance video analytics. IEEE Trans Cloud Comput. 2016;7:1152–67.
42. Vrochidis A, et al. A deep learning framework for monitoring audience engagement in online video events. Int J Comput Intell Syst. 2024;17:124.
43. Do TL, et al. Event retrieval from large video collection in ho chi minh city ai challenge 2024. In: International symposium on information and communication technology. 2025; 3–17.
44. Liu W, et al. Automatic text summarization method based on improved textrank algorithm and k-means clustering. Knowl-Based Syst. 2024;287:111447.
45. Kumar S, Sharma NK, Sharma M, Agrawal N. Text extraction from images using tesseract. In: Deep learning techniques for automation and industrial applications. 2024; 1–18.
46. Sayin A, Gierl M. Using openai gpt to generate reading comprehension items. Educ Meas Issues Pract. 2024;43:5–18.
47. Aleksic PS, et al. Bringing contextual information to google speech recognition. In: Interspeech. 2015; 468–472.
48. Gao Y, Srivastava BML, Salsman J. Spoken english intelligibility remediation with pocketsphinx alignment and feature extraction improves substantially over the state of the art. In: 2018 2nd IEEE advanced information management, communicates, electronic and automation control conference (IMCEC). 2018; 924–927.
49. Trabelsi A, Warichet S, Aajaoun Y, Soussilane S. Evaluation of the efficiency of state-of-the-art speech recognition engines. Procedia Comput Sci. 2022;207:2242–52.
50. Wang S, Yang CH, Wu J, Zhang C. Can whisper perform speech-based in-context learning? In: ICASSP 2024-2024 IEEE international conference on acoustics, speech and signal processing (ICASSP). 2024;13421–13425.

51. Patience OO, Amaechi EM, George O, Isaac ON. Enhanced text recognition in images using tesseract OCR within the laravel framework. Asian J Res Comput Sci. 2024;17:58–69.
52. Zhuravlev AA, Aksyonov KA. Dependence comparison of the effectivness of software tools for splitting video into frames on format and resolution using a road survey as an example. In: 2024 International Russian automation conference (RusAutoCon). 2024; 468–472.
53. Auger T, Saroyan E. Overview of the openai apis. Generative AI for web development: building web applications powered by OpenAI APIs and Next.js. 2024; 87–116. https://doi.org/10.1007/979-8-8688-0885-2_6.
54. Giannopoulos D, Papaioannou P, Ntzogani L, Tranoris C, Denazis S. A holistic approach for 5g network slice monitoring. In: 2021 IEEE international mediterranean conference on communications and networking (MeditCom). 2021; 240–245.
55. Patras5G. Patras5g testbed. https://wiki.patras5g.eu 2025; Accessed: 30-06-2025.

Publisher's Note