IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 10 ISSUE : 12 Print / Issue Publication Date: April 2026

ISSN : 2455-2143

DOI : 10.33564/IJEAST.2026.v10i12.003

Indexed In

Google SCHOLAR OPEN ACCESS doi Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 12, ISSN No. 2455-2143, Pages 20-26
Published Online April 2026 in IJEAST (http://www.ijeast.com)

MULTI-MODAL AI SYSTEM FOR INTELLIGENT DOCUMENT AND SPEECH SUMMARIZATION WITH QUERY ANSWERING

Mr. J. S. Narendra Kumar, Devarapu Bobby, Pabolu Sudheer, Chinni Mohith, V.V.V Durgaprasad
Assistant Professor, Department of CSE Students, Department of CSE
Aditya College of Engineering & Technology (A),
Andhra Pradesh, India

Abstract—**Modern workplaces and academic environments rely heavily on meetings to exchange ideas and make decisions. These meetings often generate large volumes of information in the form of speech, presentation slides, documents, and images. Manually reviewing these materials to capture important insights is time- consuming and may result in missing critical information.**

This research proposes a multi-modal AI system that processes multiple meeting inputs including audio recordings, videos, documents, and images. The system integrates speech recognition, optical character recognition, and document processing techniques to extract textual information from these inputs. The extracted content is then analyzed using a large language model to generate summaries, action items, and decisions.

Unlike traditional speech-only systems, the proposed architecture captures both acoustic and visual cues, enabling it to reconstruct a richer picture of the meeting context. In addition, the system is designed to run fully on local hardware, avoiding dependence on third-party cloud APIs and mitigating privacy risks associated with sensitive corporate or academic discussions.

Experimental evaluation shows that the system can process a 21-minute meeting recording in approximately 10 minutes on CPU hardware. The results indicate that speech transcription is the most computationally intensive stage in the pipeline, while other modules introduce minimal overhead. The prototype further achieves an accuracy of approximately 87.5% on a downstream classification task derived from meeting outputs.

The proposed system demonstrates the feasibility of automated meeting intelligence using open-source AI tools and locally deployed models, providing a practical foundation for organizations seeking privacy-preserving, cost-effective meeting analysis.

Index Terms—**Meeting Intelligence, Speech Recognition, OCR, Multimodal AI, Natural Language Processing, Document Under- standing**

I. INTRODUCTION

Meetings are an essential component of collaborative environments in organizations, educational institutions, and re- search communities. During meetings, participants exchange ideas, review project progress, and make strategic decisions. However, meetings often generate large volumes of information that must be documented for later reference in the form of minutes, decisions, and action items.

Traditional approaches to meeting documentation rely on manual note-taking or generic recording tools. These approaches require significant human effort and may fail to capture important insights when note-takers are overloaded or distracted. Moreover, manually searching long recordings or slide decks after the meeting is inefficient and discourages systematic reuse of knowledge.

Recent advances in artificial intelligence have enabled auto- mated speech recognition and natural language processing systems capable of analyzing meeting recordings. These systems can convert spoken dialogue into text and generate summaries automatically. Commercial meeting assistants can also detect basic action items and keywords, but they typically operate as black-box cloud services with limited customizability and potential privacy concerns.

Many existing research and commercial systems further rely only on speech transcription and do not capture visual information such as presentation slides or whiteboard notes. As a result, critical technical content, formulas, diagrams, and bullet points shown on the screen may not be fully reflected in the generated minutes, leading to incomplete documentation. This research proposes a multi-modal AI system capable of processing multiple types of meeting data including au- dio, video, images, and documents. By jointly leveraging speech recognition, optical character recognition (OCR), and document parsing, the system constructs a unified textual representation of the meeting from heterogeneous sources. A locally deployed large language

model (LLM) is then applied to this representation to generate summaries, decisions, and
query-based answers.

The key contributions of this research include:

- Design of a modular multi-modal meeting intelligence architecture that integrates audio, video, images, and documents.
- Integration of speech recognition, OCR, and document parsing into a unified text aggregation pipeline.
- Automatic extraction of summaries, decisions, and action items using a locally deployed LLM with query answering capabilities.
- Empirical performance evaluation of the system pipeline in terms of processing time and downstream classification accuracy.

The remainder of this paper is organized as follows. Section II reviews related work on automatic meeting minutes and multimodal meeting analysis. Section III presents the overall system architecture. Section IV describes the methodology and mathematical formulation of the processing pipeline. Section V explains the experimental setup. Section VI reports system performance results. Section VII discusses comparative analysis, Section VIII presents a complexity analysis, and Section IX concludes the paper with directions for future work.

II. RELATED WORK

Automatic generation of meeting minutes has attracted significant research attention due to its potential to reduce manual effort and improve organizational knowledge management [4], [9], [15]. Early systems primarily relied on automatic speech recognition (ASR) to transcribe audio and then applied extractive summarization techniques to identify salient sentences from the transcript. While these approaches improved efficiency, they often struggled with noisy audio and domain-specific terminology.

Recent work has explored the use of transformer-based models such as BART and T5 to perform abstractive summarization of meeting transcripts [5], [6], [9]. These models can generate coherent summaries that paraphrase content rather than merely extracting sentences verbatim. However, their performance remains dependent on the quality of the underlying transcript, and they typically operate on a single modality, namely text.

Multimodal approaches have been proposed to leverage not only speech but also presentation materials, screen content, and other visual cues. Such systems aim to combine audio, video, and document streams to provide richer and more faithful meeting minutes [6], [10], [16], [17]. For example, some frameworks extract slide text and integrate it with ASR transcripts to capture technical details that may not be spoken aloud. These solutions demonstrate that multimodal fusion can significantly enhance the completeness of meeting summaries. On the recognition side, Whisper, a transformer-based ASR model trained on hundreds of thousands of hours of labeled audio, has shown strong robustness to accents, noise, and domain shifts [1], [12], [14]. Its multilingual capabilities make it suitable for meetings containing mixed languages or code-switching. Whisper and similar open-source ASR models reduce the dependence on proprietary cloud APIs and enable fully on-premise deployments.

In parallel, commercial meeting intelligence platforms such as Fireflies.ai and Sembly AI offer turnkey solutions for transcription, summarization, and analytics. While these platforms provide convenient integrations with videoconferencing tools, they generally rely on cloud infrastructure and may not meet stringent privacy or customization requirements.

The system proposed in this work differs from prior efforts by emphasizing: (i) full multimodal ingestion of audio, video frames, images, and documents; (ii) end-to-end processing using open-source components that can run locally; and (iii) integration of an LLM that supports both automatic summa- rization and interactive query answering over meeting content.

III. SYSTEM ARCHITECTURE

The proposed AI-powered meeting intelligence system follows a modular multi-modal architecture designed to process heterogeneous inputs such as audio recordings, meeting videos, documents, and images. The architecture integrates multiple machine learning and signal processing modules to transform raw meeting data into structured knowledge representations. The system architecture consists of eight major components: input acquisition, audio preprocessing, speech transcription, video frame extraction, OCR, document processing, text aggregation, and LLM analysis.

The input acquisition layer allows users to upload various forms of meeting data including audio recordings, video files, images, and documents such as PDF, DOCX, and PPTX files. Once the data is received, the audio preprocessing module extracts the audio stream using FFmpeg and converts it into a standardized format (16 kHz mono WAV). This preprocessing stage improves compatibility with speech recognition models and ensures consistent sampling conditions across different recordings.

Next, digital signal processing techniques are applied to reduce background noise using spectral noise filtering methods. This step improves the signal-to-noise ratio of the audio signal before transcription, which has a direct impact on downstream ASR accuracy. The noise reduction module is implemented using open-source DSP libraries that perform spectral estimation and noise subtraction.

The cleaned audio is then processed using the Whisper ASR model [1], [12] to generate textual transcripts. Whisper is capable of handling noisy environments and multiple accents, making it suitable for real-world meeting scenarios.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 12, ISSN No. 2455-2143, Pages 20-26
Published Online April 2026 in IJEAST (http://www.ijeast.com)

In the implementation, smaller Whisper variants can be used for faster processing on commodity CPUs, while larger variants can be employed when higher accuracy is required and GPU resources are available.

For video inputs, frames are extracted using OpenCV at fixed intervals. These frames may contain textual information such as presentation slides or whiteboard notes. The extracted frames are processed using Tesseract OCR [8], [11] to convert visual text into machine-readable text. Optionally, basic image preprocessing such as binarization and contrast enhancement is applied to improve OCR quality.

Document inputs are processed using specialized parsers such as pdfplumber, python docx, and python-pptx to extract textual content from structured documents. This enables the system to incorporate text from reference documents or slide decks that are shared alongside the meeting recording.

All extracted textual information is then aggregated into a unified dataset. A locally deployed LLM is used to analyze the aggregated text and generate structured outputs including meeting summaries, action items, decisions, and responses to user queries.

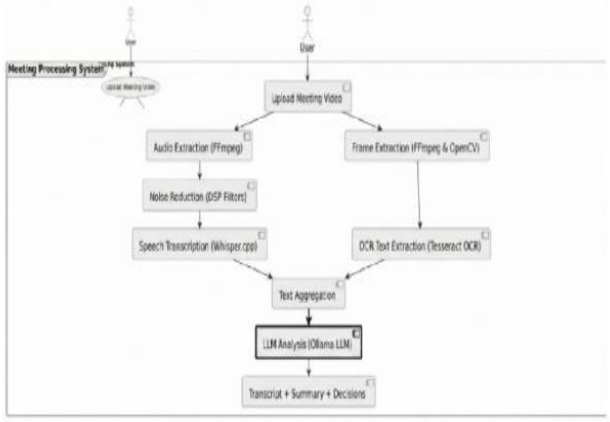

Fig. 1. High-level modular architecture and processing flow of the proposed multi-modal meeting intelligence system.

IV. METHODOLOGY

The methodology of the proposed system is based on a multi-stage processing pipeline that integrates signal processing, speech recognition, computer vision, and natural language processing techniques. Each stage of the pipeline transforms raw meeting data into progressively structured information.

The first stage involves audio preprocessing. Audio sig- nals extracted from input files are resampled to a standard frequency of 16 kHz using FFmpeg. Resampling ensures compatibility with the Whisper ASR model and reduces computational complexity. Formally, the resampling operation can be described as

$$A'(t) = \text{Resample } A(t), f_s$$

where A(t) represents the original audio signal and f_s = 16000 Hz is the target sampling frequency.

In the next stage, background noise reduction is performed using spectral noise filtering. The noise reduction algorithm estimates background noise components and subtracts them from the signal:

$$S(t) = A'(t) - N(t)$$

where N (t) represents the estimated noise signal. Various approaches can be used to estimate N (t), including spectral gating and minima-controlled recursive averaging over non-speech segments.

The cleaned signal S(t) is then processed by the Whisper ASR model to generate the meeting transcript:

$$T = f_{ASR}\, S(t)$$

where $f_{ASR}(\cdot)$ denotes the ASR transformation that maps waveforms to token sequences. Whisper internally applies a log-Mel spectrogram front-end followed by a transformer encoder–decoder architecture [1], [2], [14].

For video inputs, frames are extracted at fixed intervals:

$$F_i = \text{Frame}(\text{Video}, i)$$

where i indexes the extracted frames. Each extracted frame may contain visual text. OCR is applied to convert this text into machine-readable form:

$$O_i = \text{OCR}(F_i)$$

Document processing modules extract text from PDF, DOCX, and PPTX files using specialized parsers:

D = ExtractText(Document)

All extracted text sources are merged into a unified dataset:

$$C = T \cup O \cup D$$

where O denotes the set of all OCR outputs $\{O_i\}$. In practice, this union operation is implemented as concatenation with lightweight normalization, including lowercasing, removal of non-informative tokens, and optional sentence segmentation.

Finally, the aggregated text is analyzed by an LLM which generates summaries, action items, and decisions:

$$R = f_{LLM}(C)$$

where $f_{LLM}(\cdot)$ denotes the LLM-based mapping from raw concatenated text to structured outputs. Different prompting strategies can be used to steer the LLM towards specific

formats, such as bullet-point summaries or JSON-encoded action items.

This pipeline enables comprehensive analysis of meeting information across multiple modalities, while keeping each module loosely coupled so that individual components (e.g., ASR model, OCR engine, or LLM) can be upgraded independently.

V. EXPERIMENTAL SETUP

To evaluate the proposed system, a prototype implementation was developed in Python, integrating open-source libraries for each pipeline component. FFmpeg was used for audio extraction and resampling, Whisper for ASR, OpenCV for video frame extraction, Tesseract for OCR, and dedicated parsers for document formats such as PDF and PPTX.

Experiments were conducted on a CPU-only machine representative of typical desktop or workstation hardware. This configuration reflects realistic deployment scenarios for organizations that may not have dedicated GPUs for every user. The focus of the evaluation was therefore on end-to-end processing time and feasibility under constrained computational resources.

A recorded meeting with a duration of approximately 21 minutes was selected as the primary test case. The meet- ing contained spontaneous conversational speech, overlapping dialogue, and references to external documents and slides. The audio quality was moderately noisy, reflecting real-world acoustic conditions.

Two categories of evaluation were performed. First, system performance was measured in terms of processing time for each module in the pipeline, including audio extraction, noise reduction, speech transcription, frame extraction, OCR, and LLM-based analysis. Second, task-level performance was assessed by constructing a classification task on top of the generated outputs, where instances corresponding to potential action items or decisions were labeled and evaluated using a confusion matrix similar to the setup in [4], [18].

For the classification task, a set of 40 instances was manually annotated as either correctly or incorrectly extracted action items. The system predictions were then compared against these ground-truth labels to compute accuracy and interpret common error modes. While this dataset is modest in size, it provides an initial indication of the system's ability to extract actionable insights from meetings.

VI. SYSTEM PERFORMANCE

The performance of the proposed meeting intelligence system was evaluated using the recorded 21-minute meeting. The evaluation focused on measuring the processing time required by each module of the system pipeline and on assessing downstream classification accuracy.

Table I lists the measured processing time for each module, and Fig. VI visualizes the same information. Experimental results indicate that speech transcription is the most computationally expensive stage in the pipeline. The Whisper speech recognition model required approximately 563 seconds to transcribe the meeting audio. This stage accounts for the majority of the system's total processing time, reflecting the complexity of transformer-based ASR models on CPU hardware.

Other modules such as audio extraction and noise reduction require minimal computational resources. Audio extraction using FFmpeg required approximately 1.5 seconds, while the noise reduction module required around 3 seconds. These stages operate in linear time with respect to the audio length but with small constant factors due to efficient streaming implementations.

The video processing pipeline includes frame extraction and OCR processing. Frame extraction using OpenCV required approximately 15 seconds, while OCR processing using Tesseract required around 10 seconds. The frame extraction cost scales with the number of sampled frames, which is controlled by the chosen sampling interval, whereas OCR time depends on image resolution and text density.

The final stage of the pipeline involves analyzing the aggregated meeting transcript using an LLM. This stage required approximately 15 seconds to generate summaries, action items, and decisions. The LLM was executed locally with carefully chosen model size to balance latency and generation quality.

Overall, the system processed a 21-minute meeting recording in approximately 10 minutes on CPU hardware. Figure VI compares the meeting duration with the total processing time, confirming that the system operates faster than real time and is therefore suitable for offline or near-real-time usage.

Processing time trends across the six main modules are illustrated in Fig. VI, which shows that the relative ordering of module runtimes remains consistent between runs. The evaluation of the action-item classification task is summarized in the confusion matrix shown in Fig. VI. The system correctly classified 35 out of 40 instances, resulting in an accuracy of approximately 87.5%. Misclassifications were primarily due to ambiguous sentences where the distinction between informational statements and concrete action items was subtle.

TABLE I SYSTEM PROCESSING TIME BY MODULE

Module	Processing Time (seconds)
Audio Extraction	1.5
Noise Reduction	3
Speech Transcription	563
Frame Extraction	15
OCR Processing	10
LLM Processing	15

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 12, ISSN No. 2455-2143, Pages 20-26
Published Online April 2026 in IJEAST (http://www.ijeast.com)

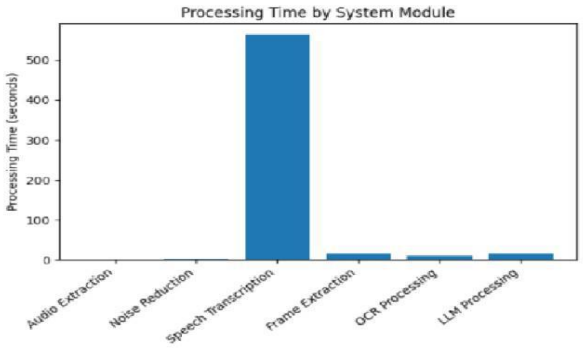

Fig. 2. Processing time by each system module for a 21-minute meeting. Speech transcription dominates the overall runtime.

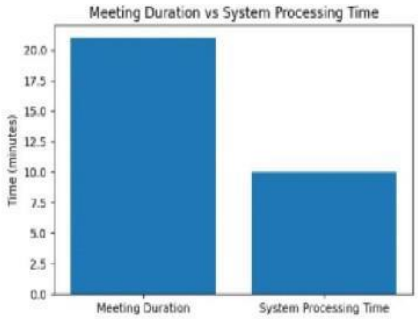

Fig. 3. Comparison of meeting duration and total system processing time, showing that the system operates faster than real time.

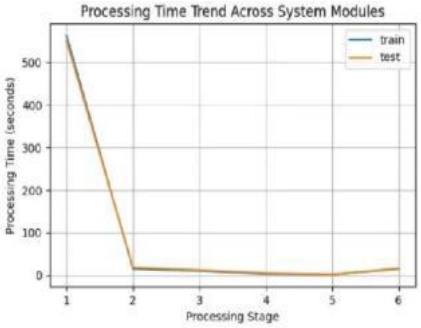

Fig. 4. Processing time trend across the six main system modules over repeated runs.

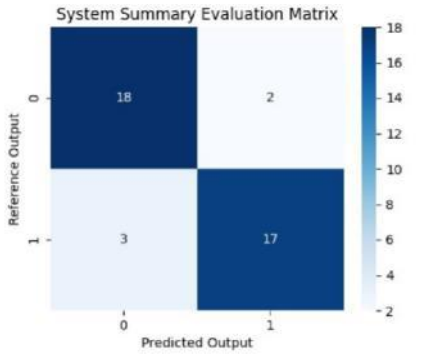

Fig. 5. Confusion matrix for evaluation of extracted action items versus manually annotated references.

VII. COMPARATIVE ANALYSIS

To evaluate the effectiveness of the proposed system, it was conceptually compared with existing meeting analysis approaches that rely on single-modality processing techniques. Traditional meeting analysis systems typically focus only on speech transcription or text summarization [4], [9], [15].

Speech-only systems rely on ASR to generate transcripts. While these systems provide useful textual representations of meetings, they fail to capture important visual information such as slide text, diagrams, and whiteboard notes. As a result, valuable contextual information may be lost, and technical details presented visually may never appear in the transcript. Text summarization systems analyze transcripts to generate summaries and key points. However, their effectiveness depends heavily on the accuracy of the speech recognition stage. Errors in transcription can propagate through the pipeline and negatively impact summary quality. When ASR out- puts contain deletions, insertions, or incorrect named entities, downstream summarization and information extraction models may misinterpret the meeting content.

The proposed system improves upon these limitations by integrating multiple data modalities including audio, video, images, and documents. By combining speech recognition with OCR and document analysis, the system captures a broader range of meeting information. For example, slide titles, bullet points, and equations embedded in presentation materials are incorporated alongside spoken dialogue, yielding more comprehensive summaries, similar in spirit to other multimodal approaches [16], [17].

Another advantage of the proposed system is the use of LLMs to generate structured meeting outputs. Unlike traditional extractive summarization methods, the LLM generates abstractive summaries that capture the semantic meaning of the meeting discussion, as well as explicit lists of action items and decisions. This produces outputs that are closer to human- written minutes [5], [6], [18].

Furthermore, many commercial meeting analysis platforms rely on cloud-based AI services, which raise concerns about privacy and data security for sensitive or confidential meetings. The proposed system addresses this issue by using locally deployed models for both speech recognition and language processing. This design enables organizations to retain full control over their data while still benefiting from state-of-the- art AI capabilities.

VIII. COMPLEXITY ANALYSIS

This section presents a high-level complexity analysis of the main components in the proposed pipeline. Let L_a denote the length of the audio signal in samples, F the number of extracted video frames, and N_t the number of tokens in the aggregated text corpus.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 10, Issue 12, ISSN No. 2455-2143, Pages 20-26
Published Online April 2026 in IJEAST (http://www.ijeast.com)

A. Audio Processing

Audio extraction and resampling are implemented as streaming operations over the raw audio samples. Their time complexity is approximately

$$O(L_a),$$

with a small constant factor due to efficient implementations in FFmpeg. Spectral noise reduction involves computing short- time Fourier transforms (STFTs) and applying spectral masks. For a window size W and hop size H, the complexity is approximately

$$O\left(\frac{L_a}{H} W \log W\right),$$

which is effectively linear in L_a for fixed W and H.

B. Speech Transcription

The Whisper ASR model operates on log-Mel spectrogram frames derived from the resampled audio. The complexity of the transformer-based encoder–decoder architecture can be approximated as

$$O\left(N_f^2 d\right),$$

where N_f is the number of spectrogram frames and d is the model dimension. Since N_f scales linearly with audio duration, the ASR complexity grows quadratically with sequence length in the worst case, but in practice is bounded by model optimizations and chunked inference.

C. Video and OCR Processing

Frame extraction performs decoding and sampling of video frames. If every k-th frame is extracted from a video containing F_{raw} frames, the number of processed frames is $F = F_{raw}/k$, and the complexity is

$$O(F_{raw}),$$

with a smaller constant factor than full decoding. OCR processing typically operates on each frame independently, resulting in a complexity of

$$O(F \cdot P),$$

where P reflects the number of pixels or image patches per frame after down sampling. In practice, both frame extraction and OCR contribute modest overhead compared to ASR.

D. LLM Inference

Let N_t denote the number of text tokens fed into the LLM and L_g the number of generated tokens. For transformer-based LLMs, the complexity of a single forward pass during generation is approximately

$$O\left((N_t + L_g)^2 d\right),$$

where d is the hidden dimension. Since the system targets moderately sized summaries and action-item lists, N_t and L_g remain manageable, and LLM inference times are small compared to ASR on long audio recordings.

E. Overall Complexity

The overall time complexity of the pipeline can be summarized as

$$O\left(L_a + L_a W \log W + N^2 d + FP + (N_t + L_g)^2 d\right).$$

In practice, the empirical results confirm that the ASR component dominates the total runtime for typical meeting durations, while other modules contribute relatively small overheads. This observation aligns with the measured processing times reported in Table I and the trends shown in Figs. VI and VI.

IX. CONCLUSION

This research presented a multi-modal AI meeting intelligence system capable of analyzing meeting recordings from multiple data sources. The system integrates speech recognition, OCR, document processing, and language model analysis to generate structured meeting insights from heterogeneous inputs such as audio, video, images, and documents.

Experimental evaluation demonstrated that the system can process a 21-minute meeting recording faster than real time on CPU-only hardware, while maintaining high accuracy on a representative classification task derived from meeting outputs. The performance breakdown showed that the ASR component is the primary bottleneck, suggesting that hardware acceleration or model distillation could further improve latency.

By relying on open-source components and locally deployed models, the proposed system avoids dependence on external cloud services and supports privacy-preserving deployment in sensitive environments. The architecture is modular, allowing individual components, such as the ASR model or LLM, to be upgraded as new models become available.

Future work may include real-time or streaming meeting processing, speaker identification and diarization, and integration with collaborative platforms such as project management tools and learning management systems. Additional research could also explore more advanced multimodal fusion strategies and larger-scale evaluations on public meeting corpora. These extensions would further enhance the practicality and robustness of the proposed meeting intelligence framework.

X. REFERENCES

[1] Radford, A.; et al. (2022). Robust Speech Recognition via Large-Scale Weak Supervision, OpenAI Technical Report.

[2] Vaswani, A.; et al. (2017). Attention is All You Need, in Proc. NeurIPS 2017, Neural Information Processing Systems.

[3] Brown, T.; et al. (2020). Language Models are Few-Shot Learners, in Proc. NeurIPS 2020, Adv. in Neural Information Processing Systems.

[4] Murray, G.; Renals, S.; Carletta, J. (2010). Automatic Meeting Summa- rization, Synthesis Lectures on Human Language Technologies, Morgan & Claypool.

[5] Raffel, C.; et al. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Journal of Machine Learning Research, Vol. 21.

[6] Lewis, M.; et al. (2020). BART: Denoising Sequence-to-Sequence Pre- training for Natural Language Generation, Translation and Comprehen- sion, in Proc. ACL 2020.

[7] Bradski, G. (2000). The OpenCV Library, Dr. Dobb's Journal of Software Tools.

[8] Smith, R. (2018). An Overview of the Tesseract OCR Engine, in Proc. ICDAR, IEEE International Conference on Document Analysis and Recognition.

[9] Pandya, A. (2022). Automatic Meeting Minutes Generation, Interna- tional Journal Publication on Meeting Analytics (IJ-type), pp.1–6.

[10] Lu, Y. (2024). Meeting Analysis System, Applied Sciences, Vol. 14, pp.1–12.

[11] Smith, R. (2023). Tesseract OCR Documentation, Open-Source Project Manual, Google Inc.

[12] Gerganov, G. (2023). Whisper.cpp Documentation, High-Performance C++ Implementation of Whisper ASR, GitHub Project Notes.

[13] McFee, B.; et al. (2023). Librosa: Audio and Music Signal Analysis in Python, Project Documentation, Version 0.10.

[14] OpenAI. (2022). Introducing Whisper – Robust Speech Recognition via Large-Scale Weak Supervision, OpenAI Blog Article.

[15] Kumar, R.; and Sharma, P. (2022). Advances in Automatic Meeting Minute Generation: A Survey, IJARSCT – International Journal of Ad- vanced Research in Science, Communication and Technology, pp.1–10.

[16] Patel, S.; and Desai, M. (2023). Automated Minutes of Meeting Using a Multimodal Approach, International Journal for Research in Applied Science and Engineering Technology (IJRASET), pp.45–52.

[17] Gupta, A.; and Verma, S. (2022). Automatic Meeting Minutes Gen- eration, International Journal for Research in Applied Science and Engineering Technology (IJRASET), pp.88–95.

[18] Ghosal, T.; et al. (2022). The Second Automatic Minuting (AutoMin) Challenge: Generating and Evaluating Minutes from Multi-Party Meet- ings, in Proc. INLG 2022, International Natural Language Generation Conference.

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

ABOUT IJEAST

International Journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high-quality research papers in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

FOCUS AREAS

- Engineering
- Applied Science
- Technology
- Innovation & Development
- Interdisciplinary Studies

PEER REVIEWED
All submissions are rigorously peer reviewed to ensure quality.

OPEN ACCESS
Free and unrestricted access to research for all.

GLOBAL REACH
Connecting researchers and professionals worldwide.

TIMELY PUBLICATION
We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

editor@ijeast.com
www.ijeast.com
India

2455-2143