








SyncFlow: A Scalable Platform for Multimodal Learning Analytics

Umesh Timalsina* , Eduardo Davalos* , Nihar Purshottam Sanda ,
Yike Zhang , Joyce Horn Fonteles , T S Ashwin , and Gautam Biswas 
Vanderbilt University, USA
{umesh.timalsina, eduardo.davalos.anaya, nihar.purshottam.sanda,
yike.zhang, joyce.horn.fonteles, ashwin.tudur.sadashiva, gautam.biswas}@vanderbilt.edu

Abstract—The new wave of educational technologies (EdTech) is revolutionizing digital education but faces challenges with the complexities of multimodal human interactions in computer-based learning environments (CBLEs). Researchers are investigating multimodal learning analytics (MMLA) as a comprehensive approach to analyzing and supporting students. However, the integration of MMLA into scalable and automated learning environments is difficult because of the absence of standardized solutions for reliable multimodal data collection and analysis. Current MMLA systems are limited in their compatibility with modern web technologies and infrastructure for browser and Internet-of-Things (IoT) integration. To address these challenges, we introduce SyncFlow, an open-source platform offering scalable, robust cloud infrastructure for automated MMLA deployments. This paper presents an end-to-end application of SyncFlow, demonstrating its integration with AI-powered CBLEs and illustrating its capabilities. SyncFlow bridges critical gaps in MMLA data collection and processing, supporting scalable and impactful CBLEs in real-world settings.

Index Terms—MMLA, streaming, platform, multimedia, multimodal data collection, real-time, artificial intelligence, data pipelines, learning environments

I. INTRODUCTION

Advances in educational technologies (EdTech) are transforming learning and teaching. Computer-based learning environments (CBLEs) create spaces for student-teacher engagement in the learning process. Traditionally dependent on unimodal log data like clickstreams, CBLEs provide limited insights into the multimodal nature of learning. The next generation of CBLEs aims to deliver immersive, adaptive, and personalized education using **Multimodal Learning Analytics (MMLA)** powered by AI algorithms [1], [2]. A significant advancement in MMLA-enhanced CBLEs is the ability to collect and analyze multimodal data in classroom settings, offering a holistic understanding of students' learning behaviors and enhancing learning outcomes [3]–[6]. MMLA systems utilize multiple data streams (video, audio, physiological, and log files) to generate actionable analytics related to learner behaviors and performance, enabling meaningful interactions with students through engaging dialogue and visualizations. This approach is based on reciprocal **mapping** and **execution**

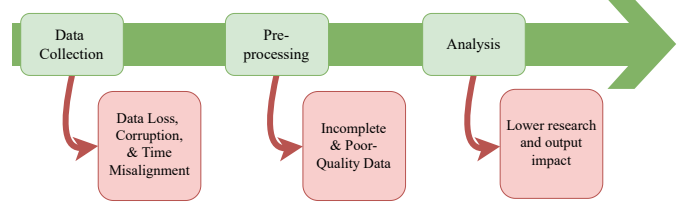


Fig. 1: **MMLA Data Management**: Manual approach to multimodal data collection, preprocessing, and analysis introduce risks that compromise data quality limiting the scale and impact of MMLA research

phases [7]. In the mapping phase, researchers correlate learning constructs with data artifacts produced by learning activities. Once mapped, MMLA systems collect, align, and analyze data streams to interpret learner activities and performance in CBLEs [5].

Enabling MMLA in CBLEs through multimodal data collection and AI-supported analyses creates a framework for comprehensive, adaptive support to learners [3], [8]. However, implementing effective MMLA systems in real-world classrooms has significant challenges [9], including:

- 1) **Technical Complexity**: Sensor heterogeneity [10] and the large volumes of multimodal data complicate data collection and processing.
- 2) **Interdisciplinary Demands**: MMLA requires synergistic interactions between developers and researchers in education, learning sciences, and AI, complicating solution development [9].
- 3) **Context Aware Automation**: Automating analytics computations in a context-aware manner that adheres to privacy and ethical concerns is challenging in classroom environments [11].

These challenges impact three key areas: (1) Efficient and scalable data collection from diverse sensors [6], [12] (2) Seamless data fusion across modalities, and (3) Deploying *end-to-end systems* that are scalable, sustainable, and ready for real-world classroom use without CBLE personnel intervention [13]. Addressing these challenges is essential for the success of MMLA deployment in CBLEs [11].

Current MMLA research and integration efforts are one-off, generating tailored solutions that are difficult to standardize,

This work has been supported by the National Science Foundation (NSF) under grant number DRL2112635.

*Equal Contribution

TABLE I: **Feature Table of Existing MMLA Collection Systems:** six current MMLA collection systems provide different advantages/disadvantages that further impact the applicability to a CBLE.

Solution	Browser Integration	IoT Integration	Multi Device Sync	Time Alignment	AI Inference	Scalability	License
iMotions ¹	△	✓		✓	✓		Proprietary
PSI [14]		✓	✓	✓	✓		MIT
SSI [15]		✓	✓	✓	✓		Unknown
LearningHub [16]		✓	✓	✓			Unknown
EZ-MMLA [17]	✓				✓	✓	Unknown
ChimeraPy [18]		✓	✓	✓	✓		GPLv3
SyncFlow	✓	✓	✓	✓	✓	✓	Apache 2.0

impacting their reusability and scalability [16]. Thus, cutting-edge MMLA is rarely deployed in classrooms [13]. Moreover, the lack of standardization requires manual effort and reduces reliability, diminishing research power [4].

The widespread adoption of web technologies has accentuated deployment challenges in classrooms. As cloud computing and modern data infrastructures evolve, deployment of MMLA in CBLEs remains uncertain. High computational demands of multimedia sources like video, which are resource-intensive to stream, process, and analyze, further complicate this. Solutions must balance cloud-based platforms' scalability with real-time multimedia data handling performance needs, where significant gaps exist.

To address these challenges, we propose **SyncFlow**², a scalable, cloud-first platform unifying multimodal data collection, processing, and feedback. Integrating technological stacks popular in video conferencing applications [19] and scalable Machine Learning (ML) deployments with novel components we developed, **SyncFlow** seamlessly integrates browser-based applications, IoT devices, and real-time AI pipelines. This unified platform bridges critical scalability, interoperability, and real-time feedback gaps within MMLA-powered CBLEs, enabling automated workflows.

We leverage **SyncFlow** for multimodal data collection and analysis, making the following key contributions:

- 1) Developing **SyncFlow**, an open-source platform and ecosystem for multimodal data collection, processing, archival, and feedback tailored for MMLA research and MMLA-powered CBLEs
- 2) Introducing a novel solution for seamless integration of IoT devices and web browsers for MMLA-powered CBLEs, providing a unified hardware and software integration platform.

II. LITERATURE REVIEW

The potential for using multimodal data to generate valuable insights in educational settings is well-recognized [3], [8]. Effective observation, tracking, and analysis of multimodal data empower CBLEs to provide evidence-based insights, deliver rich visualizations, and make better decisions. However, the seamless collection of multimodal data presents substantial technological challenges and costly software engineering efforts. A 2022 literature review [13] found that none of the 96 MMLA studies surveyed used systems capable of generating

learning analytics automatically and in real-time. The inability to deliver analytics during active teaching and learning greatly diminishes their potential deployment in classrooms, limiting their real-world impact.

High costs create substantial entry barriers, leading many MMLA-powered CBLE systems to rely on off-the-shelf tools like Open Broadcaster Software (OBS), Camtasia, log listeners, and custom scripts for data capture [20]. While practical, these ad hoc solutions face operational challenges, such as an overreliance on human technical expertise during operation. This increases the risks for data loss, corruption, degradation, and time misalignment [21]. Additionally, the need for MMLA researchers and IT support to oversee data collection limits scalability and practicality in real-world settings. These challenges diminish the amount and quality of useful multimodal data that can be collected for analysis, thus undermining the impact of MMLA research [4], [21].

Table I shows several solutions have been proposed to address these technical challenges, with iMotions¹ being widely used in MMLA research due to its extensive suite of data collection and processing tools. However, iMotions is proprietary, requires expensive license fees for each device deployed, restricts the types of analyses that can be performed, and lacks support for providing online feedback within host applications. These drawbacks make iMotions a less scalable and cost-effective option for MMLA systems.

In response, the research community has developed several open-source alternatives, including Microsoft's Platform for Situated Intelligence (PSI) [14], Social Signal Interpretation (SSI) [15], Multimodal LearningHub [16], and ChimeraPy [18]. These frameworks address key challenges, including recording local IoT devices, ensuring time alignment across multiple devices and streams, and incorporating AI pipelines into their toolkits. They enable data collection from unconventional sensors, including eye-tracking glasses, depth cameras, and HoloLens, expanding possibilities for multimodal research.

The recent shift in CBLE and supporting MMLA research from desktop to web applications has been driven by the scalability, ease of development, and streamlined deployment offered by web platforms [17]. Cloud infrastructures have enabled these applications to collect large amounts of data and perform analytics at unprecedented levels. To ensure user privacy and compliance with modern web standards, these

²<https://github.com/oee-isis-vanderbilt/syncflow.git>

¹<https://imotions.com>

applications communicate via encrypted protocols such as HTTPS and WSS, adhering to browser security specifications. In contrast, many existing IoT integration solutions rely on non-encrypted channels (e.g., non-TLS MQTT, HTTP, and WS) due to their local-first design, making them incompatible with the modern web tech stack. While web-focused MMLA tools like EZ-MMLA [17] and MediaPipe [22] offer robust multimodal data analysis capabilities, they lack mechanisms for time alignment and synchronization across multiple devices. Consequently, there is no *de facto* open-source solution for collecting and managing multimodal data tailored to web-based CBLEs.

Seeking a more web-friendly approach, we found that cloud-based conferencing systems like Zoom, Skype, and Microsoft Teams are better suited for MMLA collection systems, including multi-device (human and bots) and multi-stream time alignment and streaming [23]. However, these proprietary systems lack the flexibility to integrate with web-based CBLEs. Our search for open-source and SDK-available solutions led us to the LiveKit [19] ecosystem, which offers essential infrastructure and tools such as a Selective Forwarding Unit (SFU) media server, AI agents [24], Egress configurations, and multi-language support. **SyncFlow** builds on LiveKit to create a comprehensive MMLA solution by integrating IoT devices with its architecture, delivering a complete toolkit for AI-driven processing and feedback in MMLA applications.

III. SYNCFLOW PLATFORM

Previous work on multimedia data collection, processing, and synchronization has primarily relied on systems programming languages – constraining their ability to capture browser data. While these methodologies are effective for localized, one-off studies, the development of **SyncFlow** was guided by the following objectives:

- **Ensure seamless usability** across web browsers and IoT devices with systems programming language support; we employ *Rust*² to achieve high efficiency with our IoT and network drivers.
- **Adopt open-source development principles**, i.e., our GitHub repository is made public and under an Apache 2.0 license, and our source code is freely accessible for viewing, use, and modification. The accompanying documents promote transparency and are available for peer review.
- **Reduce human effort** in collecting data from various devices and applications; our automated data collection systems integrate heterogeneous IoT devices for seamless data synchronization, processing, and storage; in addition, they integrate cloud-based services into the data collection and analysis framework.
- **Use available tools** where possible to avoid reinventing the wheel; e.g., we use the LiveKit infrastructure for data collection and synchronization, and existing deep learning models are fine-tuned to support analytics generation.

The **SyncFlow** platform, illustrated in Fig. 2, integrates modular components to optimize session-based multimodal data streaming for MMLA systems. At its core, **SyncFlow** organizes activities into secure HTTPS/TLS sessions, which serve as the fundamental units for capturing, synchronizing, and processing multimodal data. Leveraging LiveKit’s SFU media server, **SyncFlow** ensures low latency, multi-device synchronization, and real-time data streaming across diverse sources, including browser-based clients and IoT devices. While LiveKit’s Egress servers handle multimedia data extraction, **SyncFlow** extends this capability with a custom data message extraction plugin, enabling the seamless collection of non-multimedia streams such as system logs and chat messages. This integration ensures comprehensive data coverage across modalities and aligns all data streams within the unified timeline.

SyncFlow addresses the current challenges in building MMLA systems for CBLEs by providing several *functional components*, each supporting a critical aspect of the MMLA workflow, including data collection, processing, and feedback generation. While **SyncFlow** leverages existing technologies such as LiveKit, RabbitMQ, Kubernetes, and SheldonCore for underlying services like media streaming, message brokering, and container orchestration, its primary contribution lies in integrating these components into a unified, cohesive platform. This integration streamlines the development of MMLA-powered systems by providing a scalable and modular architecture that reduces the complexity of deploying and managing multimodal workflows. By offering tailored solutions such as IoT device integration, real-time AI pipelines, and comprehensive data storage and visualization tools, **SyncFlow** standardizes and simplifies the development of MMLA-powered CBLEs, enabling researchers and developers to focus on innovation rather than infrastructure.

A. Data Collection and CBLE Integration

SyncFlow integrates seamlessly with CBLEs through a project-based architecture, where each project begins with resource provisioning by the project owner. This flexibility allows owners to manage hardware configurations and streaming requirements tailored to the scale needs of their integration. Using multi-language clients (Python, NodeJS, and Rust), **SyncFlow** enables automated synchronization of browser-based applications and IoT devices, ensuring centralized, real-time multimodal data capture.

To streamline IoT integration, **SyncFlow** organizes devices into **groups** based on their physical locations (e.g., classrooms) and orchestrates automatic session participation using RabbitMQ. Devices such as wall-mounted cameras and microphones join sessions seamlessly, reducing manual setup efforts. The **SyncFlow** dashboard further enhances usability by allowing users to dynamically record, stop, and monitor streams in real-time, ensuring adaptable and efficient data collection workflows.

²<https://www.rust-lang.org/tools/install>

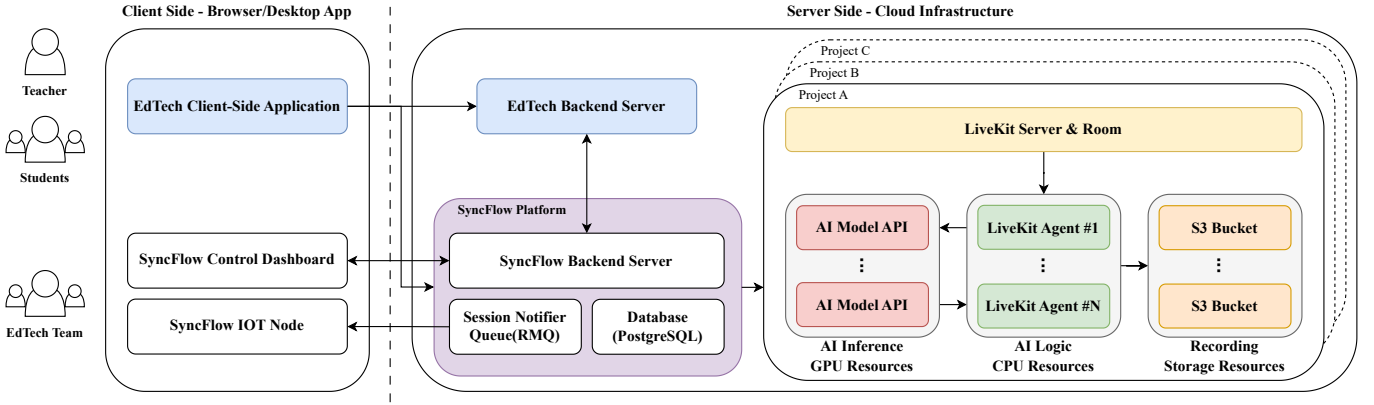


Fig. 2: **SyncFlow Platform Architecture**: The architecture consists of modular components enabling efficient resource management and orchestration for MMLA applications. Each project operates as an isolated unit, incorporating CBLE clients, local devices and sensors, and LiveKit servers for streaming. The platform captures an educational activity as a project session, provides multi-language client libraries for integration, and utilizes LiveKit egress for multimedia recording, storing data in project-specific S3 buckets.

B. Data Processing and Fusion

Given the heterogeneity of sensors and human actors present in MMLA-powered CBLEs [10], **SyncFlow** employs a unified timeline to align all collected modalities and artifacts to a common temporal reference, integrating diverse data sources like video recordings and system logs for richer insights.

For real-time processing, **SyncFlow** configures AI-driven computational pipelines using LiveKit Agents [24], stateful bots that join sessions to automate workflows by handling media tracks. Compute-intensive tasks, such as state-of-the-art AI model inference, are offloaded to Kubernetes-managed stateless entry points powered by Seldon Core³, ensuring scalable workload distribution and minimizing latency.

This architecture supports scalable AI processing and flexible resource management, allowing developers to meet project-specific CPU and GPU requirements. By offloading GPU-intensive tasks to dedicated processing infrastructure, developers can integrate third-party cloud solutions as needed, expanding compute resources for MMLA analysis. Consequently, instant feedback becomes a reality, with MMLA insights immediately available in the CBLE.

C. Archival, Feedback, and Visualization

By integrating automated streaming, synchronization, and AI inferencing, **SyncFlow** provides real-time MMLA recording and feedback. Insights from AI pipelines support proactive interventions within sessions, while captured data and analytics are stored in project-specific S3 buckets for post-processing, such as generating reports and dashboards.

These components collectively position **SyncFlow** as a versatile and scalable solution for MMLA-powered CBLEs. By seamlessly integrating data collection, real-time processing, and feedback while offering flexibility in resource management and deployment, **SyncFlow** addresses the diverse needs

of modern educational environments. It lays a strong foundation for scalable, data-driven learning technologies grounded in years of MMLA research.

IV. CASE STUDY: ONLINE AFFECT ANALYTICS AND REFLECTION TIMELINE TOOL

We demonstrate the utility of **SyncFlow** using a case study to highlight how its components were deployed in a narrative-centered learning environment to study collaboration among students. The integration showcases **SyncFlow**'s ability to automate multimedia data collection and online analysis to populate a reflection timeline [25] for post-session analysis of students' learning behaviors by the research team.

The study involved 20-24 grade six students organized into groups of 3-4 students each, seated at rectangular tables designed to accommodate individual laptops. Students collaborated on a game-based quest using speech and the game's chat interface, with their laptop webcams, microphones, and gameplay logs streaming upon login using **SyncFlow**'s real-time session orchestration. Classroom cameras and microphones, activated via **SyncFlow**'s IoT infrastructure, joined the sessions simultaneously via **SyncFlow**'s IoT infrastructure, joined the session simultaneously to capture group interactions and classroom-wide dynamics during the 45-60 minute sessions. Researchers monitored 70-90 multimodal data streams through **SyncFlow**'s dashboard (Figure 3 [left]). The study ran for 5 days, with one session conducted per day. Over 5 days, a total of 418 files were processed, consisting of 161 audio files (3.4 GB), 154 video files (28.8 GB), and 103 log files (31.8 MB), with a combined total size of 32.2 GB.

Data collection and processing were efficient, capturing multimodal streams that included participant video, audio, gameplay logs, and classroom-wide interactions, all aligned using **SyncFlow**'s unified timeline. Table II provides an overview of the processing metrics, demonstrating that data

³<https://github.com/SeldonIO/seldon-core>

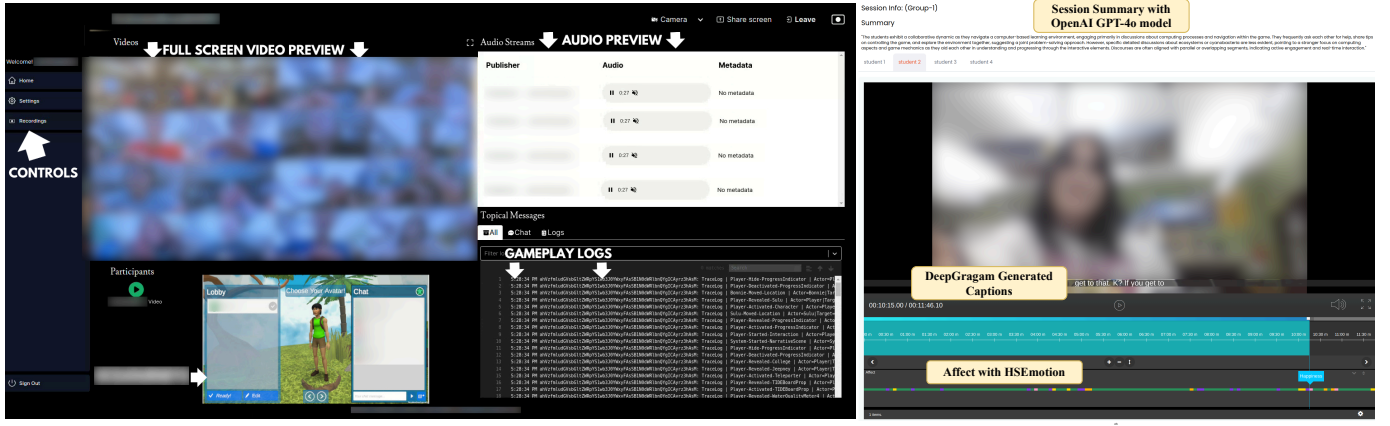


Fig. 3: **SyncFlow Live-Session Data Dashboard**: to inspect, control, and administrate the session and its data streams (left). **Reflection Timeline Dashboard for the Case Study**: to be used by learning scientists after the completion of a session (right).

streams, including near real-time transcription and affect detection, were processed within 5-6 minutes of session completion⁴. Initial challenges included stitching audio with video and handling multiple files generated by students logging out and back in during sessions. As the study progressed, we addressed these challenges by optimizing the pipeline and adding additional microphones and cameras to improve group and classroom data capture.

analyze participants' videos and compute their affective states. Combined with OpenAI's GPT-4-based session summarizer assistant, where we used transcripts and the gameplay logs, the processed artifacts provided detailed insights into student interactions. These outputs were integrated into a multimedia reflection timeline, illustrated in Figure 3 (right), allowing learning scientists to explore collaborative behaviors post-session. The architecture of the overall computation process is shown in Figure 4.

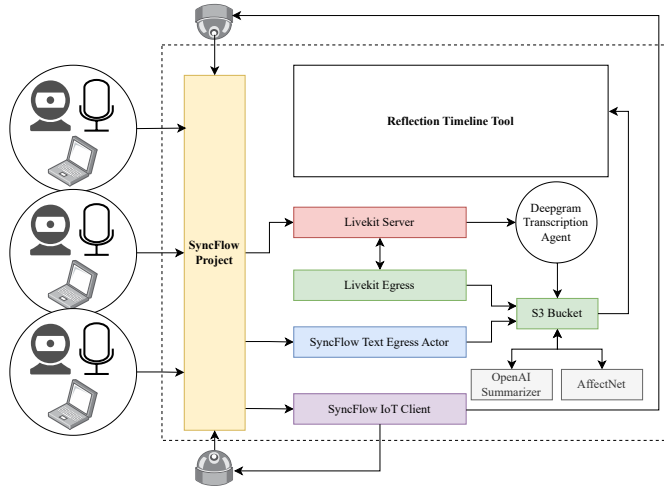


Fig. 4: Illustration of the application architecture and its integration with **SyncFlow** components.

Real-time transcripts were generated for each participant's conversation using Deepgram's⁵ transcription service, integrated via **SyncFlow**'s computational infrastructure (LiveKit Agent). These transcripts were streamed and stored in a project-specific S3 bucket, ensuring efficient data persistence. Upon session completion, HSEmotion [26] was employed to

⁴In the future, the computation will be online and information will be provided to chatbots in the environment to scaffold student learning.

⁵<https://deepgram.com>

TABLE II: Data Collection/Processing Metrics. For stitching audio with the videos collected, a batch *ffmpeg* job was run with GNU *parallel* (4 workers) per group and we report the total time taken; 5 identical HSEmotion models processed the stitched videos in parallel, followed by the GPT Assistant call, subsequently making the artifacts for the timeline tool ready

Day	Stitching Time	Affect Generation Time	GPT Call Time	Time Till Timeline Ready
I	31.612 secs	188 secs	29.027 secs	~ 5 mins
II	40.305 secs	241 secs	30.005 secs	~ 6 mins
III	39.820 secs	230 secs	23.600 secs	~ 5 mins
IV	42.303 secs	235 secs	25.333 secs	~ 6 mins
V	38.850 secs	240 secs	24.725 secs	~ 6 mins

V. DISCUSSION AND IMPLICATIONS

The case study in Section IV highlights the reliability and flexibility of **SyncFlow** in overcoming the challenges of multimodal data collection, processing, and analysis in real educational settings. Over the five days, **SyncFlow** securely gathered roughly 32GB of data. It employed its timeline-based fusion to synchronize audio, video, and gameplay logs for online and post-hoc analyses. The reflection timeline tool notably benefited from this integrated pipeline, allowing learning scientists to investigate collaborative behaviors alongside

automated artifacts, including transcriptions and affect detection. With numerous heterogeneous systems operating independently yet streaming data into one framework, maintaining a consistent temporal reference across diverse modalities made data fusion remarkably straightforward.

Below, we outline the impact of **SyncFlow** on key stakeholders involved in the development and use of CBLEs:

- **For Learning Scientists & Analysts:** **SyncFlow** streamlines real time MMLA data collection and management, reducing manual effort and risks of data loss. It enhances data reliability and bridges research and practice by providing tools for online actionable feedback and real-world visualizations.
- **For CBLE Developers:** **SyncFlow** offers scalable tools to implement fully automated MMLA systems, minimizing the need for on-site intervention and enabling streamlined workflows. Its cloud-first design ensures high availability and reduces preparation efforts for MMLA studies.
- **For Students & Teachers:** **SyncFlow** empowers teachers to independently integrate MMLA tools for classroom data collection with minimal external support. Its browser-based interface ensures easy use while adhering to strict privacy protocols, enhancing accessibility for all end-users.

VI. FUTURE WORK AND CONCLUSION

SyncFlow represents a significant advance toward a unified, scalable, and secure solution for addressing key challenges in MMLA data collection and processing. By integrating browser-based applications with IoT devices through advanced cloud infrastructure, LiveKit for streaming, and RabbitMQ for IoT communication, **SyncFlow** enables automated workflows and real-time feedback in CBLE applications.

Future work will focus on enhancing the platform to improve the experience for CBLE developer, researchers, and practitioners. Key areas of development include expanding support for device and sensor data capture drivers to accommodate additional modalities and equipment. A current limitation is the manual management of CPU and GPU resources by CBLE teams to support LiveKit Agents and model API entry points. To address this, we aim to streamline and automate cloud infrastructure deployment, incorporating GitHub-based continuous integration and deployment workflows, similar to platforms like Vercel. Additionally, we plan to introduce a no-code graphical user interface to provide input to MMLA chatbots, easing the inclusion of intelligent AI within CBLEs.

We will also conduct targeted usability studies with CBLE developers and researchers/practitioners to identify *pain points* and opportunities for further enhancement. By sharing **SyncFlow** openly and fostering collaboration, we aim to democratize MMLA development, driving the adoption of accessible, data-driven CBLEs across the educational technology community.

ACKNOWLEDGMENTS

This research was supported by funding from the National Science Foundation (NSF) under Grants DRL-2112635. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

- [1] Sue Beckingham, Jenny Lawrence, Stephen Powell, and Peter Hartley, *Using Generative AI Effectively in Higher Education: Sustainable and Ethical Practices for Learning, Teaching and Assessment*, Taylor & Francis, 2024.
- [2] Anya S Evmenova, Jered Borup, and Joan Kang Shin, “Harnessing the power of generative ai to support all learners,” *TechTrends*, pp. 1–12, 2024.
- [3] Paulo Blikstein and Marcelo Worsley, “Multimodal learning analytics and education data mining: Using computational technologies to measure complex learning tasks,” *Journal of Learning Analytics*, vol. 3, no. 2, pp. 220–238, 2016.
- [4] Clayton Cohn, Eduardo Davalos, Caleb Vatrall, Joyce Horn Fonteles, Hanchen David Wang, Meiyi Ma, and Gautam Biswas, “Multimodal Methods for Analyzing Learning and Training Environments: A Systematic Literature Review,” *arXiv e-prints*, p. arXiv:2408.14491, Aug. 2024.
- [5] Daniele Di Mitri, Jan Schneider, Marcus Specht, and Hendrik Drachsler, “From signals to knowledge: A conceptual model for multimodal learning analytics,” *Journal of Computer Assisted Learning*, vol. 34, pp. 338–349, 8 2018.
- [6] Sharon Oviatt, “Ten opportunities and challenges for advancing student-centered multimodal learning analytics,” pp. 87–94, 10 2018.
- [7] Xavier Ochoa, *Multimodal Learning Analytics: Rationale, Process, Examples, and Direction*, pp. 54–65, SOLAR, 2 edition, 2022, Section: 6.
- [8] Xavier Ochoa, Charles Lang, George Siemens, Alyssa Wise, Dragan Gasevic, and Agathe Merceron, “Multimodal learning analytics-rationale, process, examples, and direction,” *The handbook of learning analytics*, pp. 54–65, 2022.
- [9] Shashi Kant Shankar, Adolfo Ruiz-Calleja, Luis P. Prieto, María Jesús Rodríguez-Triana, Pankaj Chejara, and Sandesh Tripathi, “Cimla: A modular and modifiable data preparation, organization, and fusion infrastructure to partially support the development of context-aware mmla solutions,” *JUCS - Journal of Universal Computer Science* 29(3): 265–297, vol. 29, pp. 265–297, 3 2023.
- [10] Paulo Blikstein and Marcelo Worsley, “Multimodal learning analytics and education data mining: using computational technologies to measure complex learning tasks,” *Journal of Learning Analytics*, vol. 3, no. 2, pp. 220–238, Sep. 2016.
- [11] Mutlu Cukurova, Michail Giannakos, and Roberto Martinez-Maldonado, “The promise and challenges of multimodal learning analytics,” *British Journal of Educational Technology*, vol. 51, no. 5, pp. 1441–1449, 2020.
- [12] Pankaj Chejara, Reet Kasepalu, Luis Pablo Prieto, María Jesús Rodríguez-Triana, Adolfo Ruiz-Calleja, and Shashi Kant Shankar, “Multimodal learning analytics research in the wild: Challenges and their potential solutions,” in *CrossMMLA@ LAK*, 2023, pp. 36–42.
- [13] Lixiang Yan, Linxuan Zhao, Dragan Gasevic, and Roberto Martinez-Maldonado, “Scalability, sustainability, and ethicality of multimodal learning analytics,” in *LAK22: 12th International Learning Analytics and Knowledge Conference*, 3 2022, pp. 13–23, ACM.
- [14] Dan Bohus, Sean Andrist, Ashley Feniello, Nick Saw, Mihai Jalobeanu, Patrick Sweeney, Anne Loomis Thompson, and Eric Horvitz, “Platform for situated intelligence,” 2021.
- [15] Johannes Wagner, Florian Lingenfelder, Tobias Baur, Ionut Damian, Felix Kistler, and Elisabeth André, “The social signal interpretation (ssi) framework: multimodal signal processing and recognition in real-time,” in *Proceedings of the 21st ACM International Conference on Multimedia*, New York, NY, USA, 2013, MM ’13, p. 831–834, Association for Computing Machinery.
- [16] Jan Schneider, Daniele Di Mitri, Bibeg Limbu, and Hendrik Drachsler, “Multimodal learning hub: A tool for capturing customizable multimodal learning experiences,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11082 LNCS, pp. 45–58, 2018.

- [17] Javaria Hassan, Jovin Leong, and Bertrand Schneider, "Multimodal data collection made easy: The ez-mmla toolkit: A data collection website that provides educators and researchers with easy access to multimodal data streams.," in *LAK21: 11th International Learning Analytics and Knowledge Conference*, New York, NY, USA, 2021, LAK21, p. 579–585, Association for Computing Machinery.
- [18] Eduardo Davalos, Umesh Timalsina, Yike Zhang, Jiayi Wu, Joyce Horn Fonteles, and Gautam Biswas, "Chimerapy: A scientific distributed streaming framework for real-time multimodal data retrieval and processing," in *2023 IEEE International Conference on Big Data (BigData)*, 2023, pp. 201–206.
- [19] David Zhao, "How we built a globally distributed webrtc mesh network," 10 2022.
- [20] Ran Liu and John C. Stamper, "Multimodal data collection and analysis of collaborative learning through an intelligent tutoring system," in *MMLA-CrossLAK@LAK*, 2017.
- [21] Bertrand Schneider, Richard Davis, Roberto Martinez-Maldonado, Marcelo Worsley, and Nikol Rummel, "Stepping outside the ivory tower: How can we implement multimodal learning analytics in ecological settings, and turn complex temporal data sources into actionable insights?," 06 2024, pp. 323–330.
- [22] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al., "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.
- [23] A. M. Suduc, M. Bizoi, and F. G. Filip, "Status, challenges and trends in videoconferencing platforms," *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, vol. 18, 5 2023.
- [24] Russ d'Sa, "An open source stack for real-time multimodal ai," .
- [25] Nicolas Hervé, Pierre Letessier, Mathieu Derval, and Hakim Nabi, "Amalia.js," in *Proceedings of the 23rd ACM international conference on Multimedia*. 10 2015, pp. 709–712, ACM.
- [26] Andrey V. Savchenko, "Hsemotion: High-speed emotion recognition library," *Software Impacts*, vol. 14, pp. 100433, 12 2022.