

The SOROLL-IA Framework: Overcoming the Challenges of Large-Scale Audio Database Management and Delivery for Operational Audio Tagging Systems

Jordi Grau-Haro, Javier Naranjo-Alcazar, Ruben Ribes-Serrano, Pedro Zuccarello
 Instituto Tecnológico de Informatica (ITI)
 Valencia, Spain
 {jnaranjo,jgrau,rribes,pzuccarello}@iti.es

Abstract—The operational deployment of real-world audio tagging systems introduces severe engineering bottlenecks that extend far beyond acoustic modeling, posing critical challenges in the management, versioning, and delivery of large-scale audio databases. To overcome these hurdles, this paper presents the SOROLL-IA Framework, a unified, end-to-end pipeline that governs the entire audio AI lifecycle by bridging the gap from physical edge data acquisition to production-ready audio tagging model deployment. The platform operates on a continuous, multi-stage workflow: first, distributed edge IoT nodes manage local acoustic recording and stream lightweight telemetry to a centralized infrastructure. Second, the collected audio recordings are retrieved and stored within a decoupled infrastructure, where physical file storage is separated from a relational metadata layer that indexes and orchestrates the dataset. Third, this relational layer leverages tailored active learning strategies to package and deliver informative data batches in iterative, periodic cycles to a custom spectrogram-based annotation interface, accelerating human labeling while enforcing quality control protocols. Once models are trained, they are deployed back to the edge nodes. Finally, in the production stage, the nodes transmit live inference telemetry, allowing the centralized system to monitor for domain drift and dynamically adjust subsequent data curation loops. By mapping explicit hardware constraints to microservice-driven software orchestration, SOROLL-IA establishes a scalable, production-ready blueprint that transforms volatile environmental audio data into an organized, traceable, and self-enriching database asset.

Index Terms—Audio Database, Audio tagging, IoT, active learning.

I. INTRODUCTION

Although state-of-the-art Computer Audition has made significant progress in audio tagging [1], [2] and sound event classification [3], the vast majority of contributions remain confined to static benchmark datasets, isolated model architectures, and evaluation protocols under tightly controlled, research-oriented conditions [4]–[6]. While massive repositories like AudioSet [4] have advanced theoretical modeling [7], [8], they are inherently structured for benchmarking rather than

This work was partially funded by the Valencian Institute for Business Competitiveness (IVACE). This work has been supported by Grants PID2025-174060OB-C31 and PID2022-137048OA-C44 funded by MICIU/AEI/10.13039/501100011033 and by “ERDF/EU”.

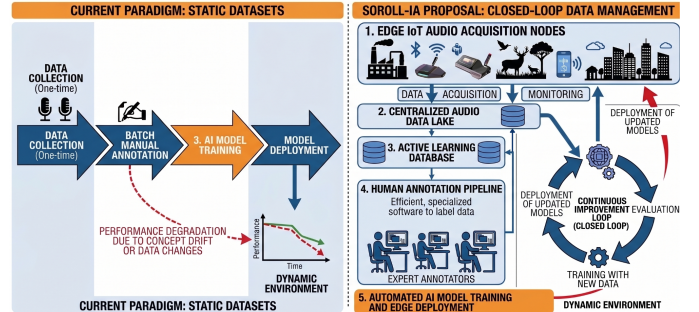


Fig. 1. Comparison between the traditional static dataset paradigm and the proposed SOROLL-IA closed-loop data management architecture.

production deployment. Industrial applications cannot rely on such pre-existing data due to severe domain shifts between controlled training environments and real-world acoustic conditions [9]. Instead, practitioners must architect, curate, and maintain domain-specific audio databases from scratch [10]. Real-world acoustic monitoring demands data infrastructures capable of handling extreme environmental volatility—from harsh mechanical noise in industrial facilities to unpredictable weather-driven variations in outdoor settings. Transitioning from research prototypes to robust deployments is thus fundamentally a challenge of building and sustaining large-scale, operationally tailored audio data lifecycles.

Traditional audio AI workflows typically treat datasets as static, one-off assets: audio is collected, manually annotated, and used to train a model before being archived. While sufficient for controlled research, this paradigm collapses under non-stationary conditions where acoustic backgrounds constantly shift and novel events emerge. Sustaining long-term accuracy therefore requires an iterative data lifecycle, where the underlying audio repository is continuously expanded, curated, and synchronized with model updates to reflect real-world conditions (see Fig 1). To address this, we shift the focus from isolated model training to end-to-end audio data engi-

neering. We present a scalable pipeline designed to manage and deliver massive audio databases—potentially encompassing millions of continuous streaming segments—and formalize the architectural requirements to transform raw environmental recordings into traceable, edge-ready tagging assets. Developed within the SOROLL-IA platform, this workflow decomposes the data lifecycle into six interconnected architectural pillars:

- 1) **Autonomous Edge IoT Acquisition:** Hardened hardware and software layers designed for continuous, privacy-compliant acoustic capture under power and network constraints.
- 2) **Decoupled Asset Architecture:** A scalable ingestion engine that decouples unstructured binary blobs from relational metadata.
- 3) **End-to-End Data Lineage Tracking:** A structured PostgreSQL database schema engineered to preserve absolute data lineage from the initial recording node to the expert annotation chunks and active learning (AL) iterations [11].
- 4) **Human Annotation Infrastructure:** A custom, spectrogram-based multi-label interface (*New Audio Annotation Tool*, or NAAT) tailored for rapid industrial taxonomy management, exporting completed annotations as structured JSON assets that are seamlessly reintegrated into the relational database.
- 5) **Continuous Model Evolution and Tracking:** An MLOps-driven workflow that triggers iterative training runs as fresh ground-truth batches are committed. It utilizes a centralized experiment registry to relationally link model versions with their exact dataset snapshots.
- 6) **Closed-Loop Model Evolution:** A production feedback pathway supported by edge inference telemetry. Instead of an isolated deployment, the system continuously streams prediction metrics and confidence scores back to the central architecture; this telemetric layer provides the baseline indicators required to detect domain drift, allowing operators to dynamically transition from foundational data-seeding strategies toward model-guided active learning approaches once the system is deployed in production.

This paper is intentionally framed as a system architecture and engineering blueprint for industrial audio data management. Rather than evaluating an isolated acoustic model under static benchmark conditions, our primary contribution lies in the formalization of the continuous, end-to-end data pipeline. The design patterns, architectural trade-offs, and lessons learned presented throughout this work stem directly from empirical knowledge acquired across multiple real-world deployments for industrial partners, focusing on outdoor and unmanaged environments for operational Audio Tagging applications. Ultimately, this framework synthesizes these field experiences into a generalizable, production-ready blueprint to sustain operational audio AI under hardware, bandwidth, and environmental constraints.

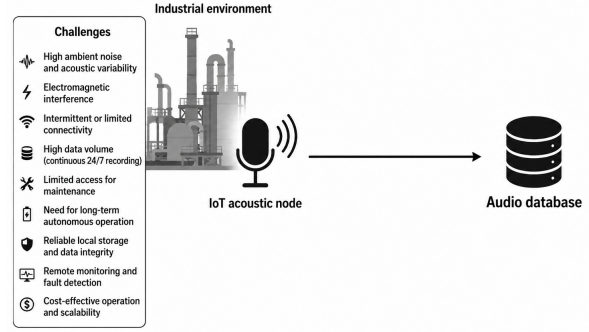


Fig. 2. Technical challenges on deployment for scalable audio acquisition in real-world environments using IoT acoustic nodes.

II. TECHNICAL CHALLENGES

A. Scalable Audio Acquisition in Industrial Environments

Implementing a continuous audio acquisition system in real-world or industrial settings introduces physical and operational challenges that go far beyond standard acoustic sensing, as illustrated in Fig. 2. Unlike controlled environments, real-world deployments dictate strict hardware and software infrastructure constraints. Equipment access is highly restricted, maintenance requires plant-wide coordination, and the acoustic environment is inherently hostile due to background noise and/or vibrations, among other considerations.

To ensure a continuous flow of high-quality data into the pipeline, the acquisition architecture must overcome three core engineering bottlenecks:

- **Node Autonomy and Edge Resilience:** Recording nodes must operate autonomously for extended periods without human intervention. This requires robust local storage strategies to prevent data loss during intermittent network outages, automated fault detection, and remote diagnostic capabilities to verify data integrity from afar.
- **Data Volume and Bandwidth Constraints:** Continuous multi-node recording generates massive data volumes that can easily saturate industrial network bandwidth. Furthermore, in many industrial settings, strict policies and privacy regulations prohibit transferring raw audio over external or shared networks to protect proprietary acoustic signatures.
- **Environmental Degradation:** Hardware components face physical degradation due to dust, temperature fluctuations, and mechanical stress.

B. Centralized Storage and Data Ingestion

Establishing a resilient link between distributed edge nodes and centralized storage demands efficient ingestion and indexing of continuous streams. As shown in Fig. 3, this architecture must overcome three key engineering challenges:

- **Exponential Volume and Scaling Limits:** High-resolution multi-node audio recording generates substantial data overhead per hour. At an industrial deployment

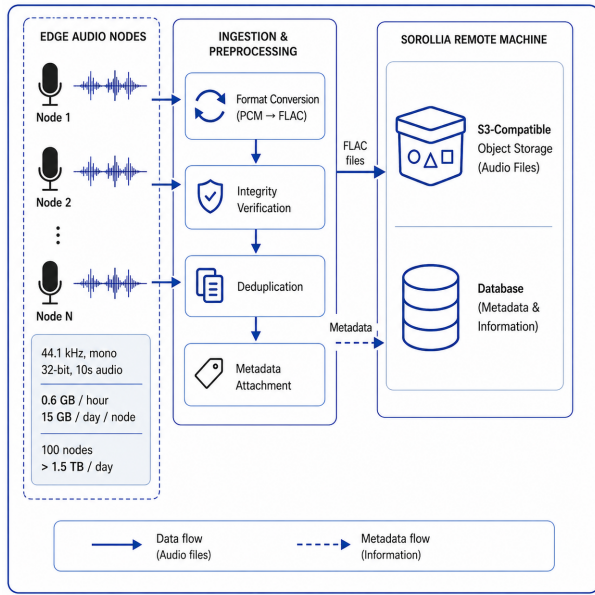


Fig. 3. Data ingestion challenges over deployment on industrial environment.

scale, this volume grows exponentially, rapidly straining storage capacity.

- **Decoupling Blob Storage from Relational Metadata:** Industrial audio is useless without its context. The system must ingest massive unstructured binary files (the raw audio) while simultaneously managing highly structured, fast-evolving metadata (timestamps, node IDs, and processing states). Designing a storage architecture that can handle petabyte-scale file lookups without losing the strict relational links required for querying constitutes a architectural bottleneck.
- **Ingestion Integrity and Lifecycle Management:** Real-world deployments require dynamic stream validation to detect duplicates and corrupted packets, alongside automated retention policies that purge redundant or low-value segments, preventing dataset contamination.

C. Structured Metadata and Annotation Management

Once the structural metadata and raw audio assets are indexed within the relational schema, the operational challenge shifts from static data storage to dynamic lifecycle management. In real-world deployments—where nodes generate millions of audio segments—human annotation is a critical bottleneck, demanding that the metadata infrastructure evolves into an active orchestrator. This operational challenge stems from two interconnected technical bottlenecks:

- **Efficient Cross-Variable Curation:** Isolating high-priority candidate recordings for human review requires filtering data across diverse, concurrent variables—such as deployment timelines, specific edge node metadata, and model confidence scores.
- **Tracking active learning Iterations:** Standard database schemas excel at cataloging static assets, but they lack

mechanisms to log the dynamic history of iterative data selection. AL workflows generate unique meta-information for each batch—such as timestamps of the selection event, target configurations, and the specific metrics that triggered a segment’s proposal.

D. Human Annotation at Scale

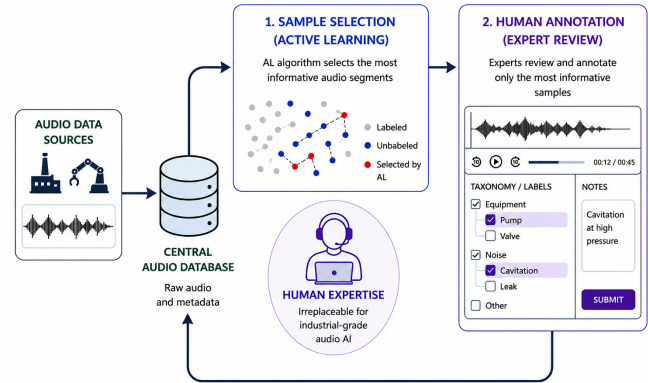


Fig. 4. Data collection challenges over deployment on industrial environment.

Despite advances in automated data curation [12], human expertise remains irreplaceable for industrial-grade audio AI [11]. AL strategies do not eliminate human operators; instead, they optimize their effort by filtering out redundancy and surfacing only the most informative audio segments.

As shown in Fig. 4, annotation forms a continuous loop tying sample selection, expert review, metadata updates, and model retraining. Scaling this workflow requires addressing three key engineering challenges:

- **Infrastructure and UX Constraints:** Specialized tools are required to handle industrial audio workflows. These tools must provide intuitive interfaces for rapid audio playback, multi-label taxonomic navigation, and efficient task management, while generating structured outputs that seamlessly sync with the central database.
- **Pipeline Throughput and Coupling:** The efficiency of the pipeline depends heavily on the fluid integration between the data selection layer (AL) and the annotation interface. While AL algorithms reduce the volume of samples needing review, their practical impact is bottlenecked without an infrastructure that dynamically serves these samples and instantly commits human feedback back into the loop.

E. AI Model Training, Evolution and Versioning

As the dataset expands through periodic (e.g., weekly) annotation batches, the machine learning models must be retrained iteratively to incorporate this new knowledge. This continuous evolution introduces a critical MLOps challenge: maintaining absolute synchronization between the data state and the model state.

Multiple training runs occur simultaneously using evolving dataset versions, shifting hyperparameter configurations, and diverse model architectures. The core engineering challenge lies in ensuring strict reproducibility and traceability—guaranteeing that any given model checkpoint can be mathematically and textually traced back to the exact training data split, the specific active learning iteration or iterations that generated it, and the precise software environment used.

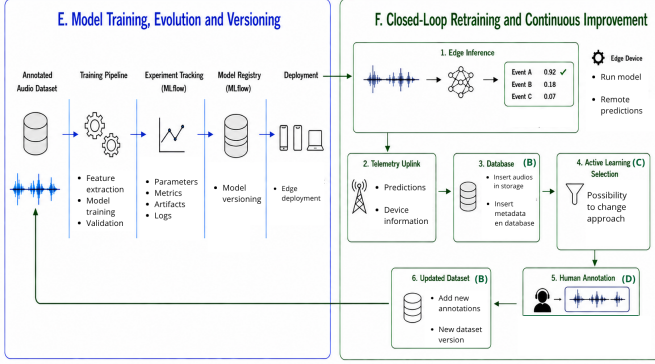


Fig. 5. Overview of the SOROLL-IA MLOps and continuous inference pipeline. Block E details the training, experiment tracking, and model registry workflow leading to edge deployment. Block F illustrates the closed-loop continuous improvement cycle, driven by edge inference telemetry, active learning updates and human annotation to update the database.

F. Closed-Loop Retraining and Continuous Improvement

The paradigm shifts fundamentally once the initial Minimum Viable Product (MVP) model is deployed into production. Industrial or open real-world environments are inherently dynamic, meaning the acoustic conditions encountered during real-time operation inevitably diverge from the static distribution of the initial training dataset. Over time, factors such as mechanical wear on machinery, environmental noise shifts, or completely unseen acoustic events trigger performance degradation, commonly known as data and domain drift [13]. Mitigating this degradation requires transforming a linear deployment pipeline into a self-sustaining, closed-loop system (see Fig. 5). The main challenge here is twofold: informational and economic. First, the infrastructure must capture and route telemetry and inference data from the edge back to the central data management layer without disrupting production bandwidth. Second, because continuous manual labeling of millions of streaming production segments is economically and operationally impossible, the system faces a selection challenge. The pipeline must intelligently identify and filter which specific production samples are truly informative or indicative of domain shift. Defining the constraints and architectural boundaries to ingest this production feedback loop seamlessly, without overloading human operators, constitutes a core bottleneck for sustaining long-term model reliability in the wild.

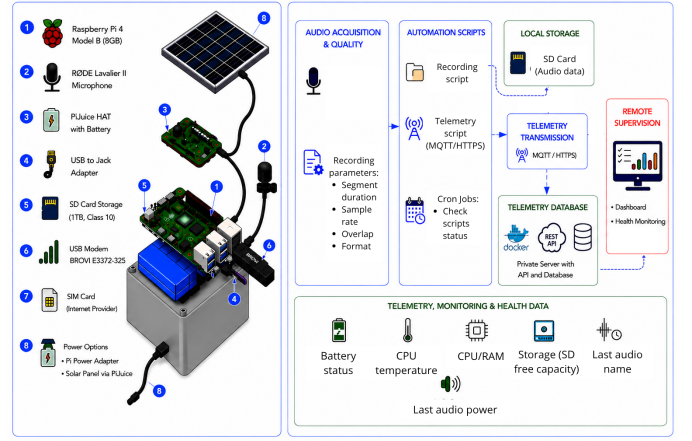


Fig. 6. Hardware and software architecture of the IoT audio node. The left side shows the physical components of the node. The right side summarizes the software stack for audio acquisition, automation scripts, telemetry, local storage and system monitoring.

III. PROPOSED FRAMEWORK AND IMPLEMENTATION

A. Edge IoT Audio Nodes

To mitigate the harsh physical and operational constraints outlined in Section II-A, we designed an autonomous, industrial-grade Edge IoT architecture. The proposed node treats hardware selection and software automation as a unified engineering solution to ensure continuous data integrity (see Fig. 6). Building upon the foundational acquisition concepts explored in our preliminary prototype [11], the mature SOROLL-IA edge node isolates and solves each specific acquisition bottleneck through distinct architectural layers.

As illustrated in Fig. 6, the deployment architecture balances computational capacity with physical hardening through a structured co-design:

- Mitigating Autonomy and Power Resilience Challenges:** To survive restricted physical access and sudden blackouts—that may occur in both industrial grids and unmanaged open environments—the node implements a versatile power management architecture. The subsystem supports a dual-input power routing scheme, allowing the node to be seamlessly powered either via a standard AC grid adapter in indoor industrial settings, or via a solar panel feed for outdoor deployments. To govern this hybrid input, we pair a Raspberry Pi 4 Model B (8 GB) with an intelligent PiJuice HAT module [14]. The PiJuice acts as a smart power regulator and backup battery: it handles safe-shutdown sequences whenever power drops below a critical threshold to prevent file system corruption, orchestrates automated boot-up sequences once current is restored, and dynamically manages the node’s power cycles based on solar availability when deployed in the wild. All the components are placed inside a IP65 waterproof box. On the software layer, this unattended autonomy is reinforced by optimized *cron* automation daemons and system services that continuously monitor system health,

verify recording states, and handle execution recovery without human intervention.

- **Bypassing Bandwidth and Privacy Constraints:** To address network saturation and strict data privacy mandates common in both urban and industrial environments, the architecture enforces an *edge-first* storage lifecycle where processing and data buffering are handled locally at the node level. High-resolution acoustic capture is routed via a RODE Lavalier II microphone [15] through a USB-to-jack converter, writing directly to an expansive 1 TB Class 10 SD card. The local recording daemon structures the acoustic stream into discrete, non-overlapping 10-second audio clips. This duration was explicitly selected to align with the standard segmentation format of established public sound repositories, such as AudioSet [4], ensuring format compatibility for downstream training while dramatically reducing network load by allowing scheduled, batch-based database synchronization. To support this *edge-first* architecture without disrupting local bandwidth constraints, the node utilizes a BROVI E3372-325 USB cellular modem [16] (equipped with an industrial SIM) dedicated strictly to low-overhead telemetry and control loops:

- **Telemetry Loop:** A lightweight automation script extracts operational metrics (battery state, CPU temperature, storage limits, and connectivity status) alongside non-invasive RMS energy estimates from the latest recorded segment to verify sensor health without transmitting raw audio and its file name.

- **Managing Environmental Degradation and Lifecycles:** Continuous telemetry observation enables predictive maintenance scheduling, allowing operators to plan interventions based on storage consumption, power fluctuations, and acoustic energy anomalies before hardware failure occurs. To ensure safe field maintenance without external network dependencies, the node dynamically hosts a local Wi-Fi access point. This permits direct laptop connectivity to the operating system, enabling graceful service termination, log inspection, and configuration validation when near the node. Following a controlled power cycle, the system automatically verifies sensor initialization and recording states upon reboot, guaranteeing operational readiness before returning to unsupervised deployment.

- **Deployment of Telemetry:** To ensure seamless integration with the SOROLL-IA data architecture, the current pipeline replaces the legacy MQTT/ThingsBoard prototyping approach [11] with a structured HTTPS-based communication stack. A lightweight telemetry agent on the edge node periodically collects operational variables and transmits them as structured JSON payloads to a dedicated REST API. The API validates requests, associates records with node identifiers, and persists them in a PostgreSQL database (Telemetry DB). On the backend, Grafana queries the database via optimized SQL to render

real-time and historical dashboards for anomaly detection, maintenance planning, and operational supervision (see Fig. 7). All backend services are containerized using Docker, ensuring reproducible deployment, service isolation, and straightforward scaling. While the initial telemetry layer focuses on hardware and acquisition monitoring, the same infrastructure is designed to incorporate future inference-related indicators (e.g., predicted classes, confidence scores, and execution latency) (see Section III-F) without architectural changes (see Fig. 8).

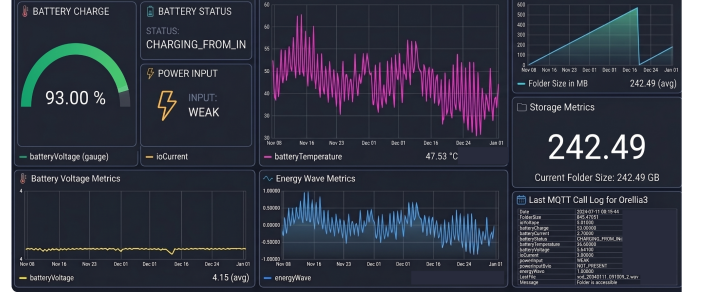


Fig. 7. Telemetry dashboard capturing real-time edge node parameters during the continuous recording phase. This centralized monitoring interface tracks CPU temperature, battery performance percentage, deployment geolocation coordinates, available SD card capacity—where sharp resets visually denote the periodic physical data retrieval events—and the root-mean-square energy profiles of the acquired acoustic streams, alongside other critical operational diagnostics to ensure field maintenance and data integrity.

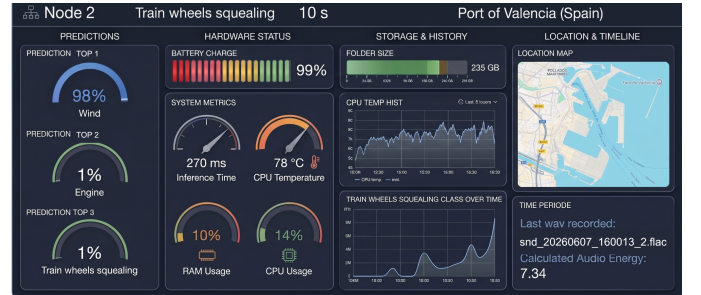


Fig. 8. Operational dashboard displaying a representative edge node deployment. The interface integrates the model’s top-3 predicted audio classes, the execution time required for the latest inference, and a historical timeline tracking a user-selected target class over time. Designed for production monitoring, this unified view tracks files logged under the standardized `snd_YYYYMMDD_hhmmss_idnode.flac` (see Fig. 10 table Nodes) naming convention alongside local hardware, storage, and thermal metrics, allowing operators to verify exact system latency and edge reliability.

B. Centralized Storage and Data Ingestion

To resolve the scalability, metadata coupling, and integrity bottlenecks described in Section II-B, the SOROLL-IA backend decouples physical object storage from relational metadata. Deployed on a centralized on-premise server using Docker (see Fig. 9), the architecture systematizes data infrastructure through three specialized implementation layers:

- **Adaptive Ingestion and Offline Preprocessing:** To mitigate network saturation over extended field operations,

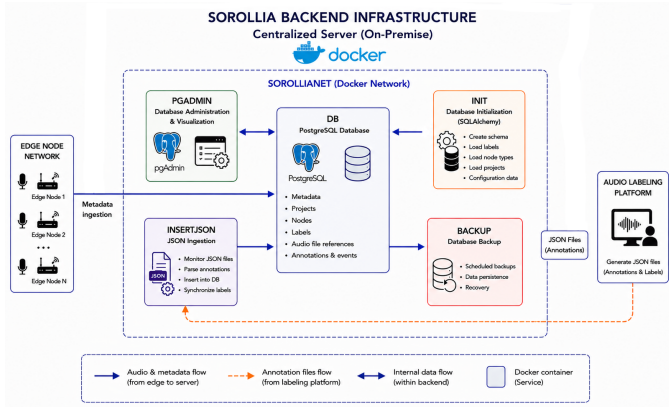


Fig. 9. Docker-based deployment architecture of the SOROLL-IA backend infrastructure hosted on a centralized on-premise server. Within the isolated SOROLL-IA network, the main PostgreSQL container (*DB*) centralizes the lifecycle registry detailed in Fig. 10. This core storage is supported by specialized utility services: the *INIT* container executes as a single-run service to instantiate relational schemas and configurations; *BACKUP* runs periodically to ensure state persistence and data recovery; and *PGADMIN* provides an interface for browser-based database administration. To ingest ground truth (see Section III-D), the *INSERTJSON* service continuously monitors a target directory via a watchdog mechanism; upon completion of an annotation batch, it parses the exported JSON files to automatically insert the new labels into the *Chunks* table (see Fig. 10), ensuring robust, containerized synchronization across the pipeline.

the platform utilizes edge telemetry metrics sent by the nodes. Operators review these centralized dashboards (Figs. 7 & 8) to track node status proactively; once a storage capacity threshold is breached, field technicians execute a physical swap of the 1 TB SD card to ensure continuous recording. Experimental trials demonstrated that uncompressed audio acquisition rapidly exhausts this local storage. To maximize deployment windows, native FLAC recording is highly recommended at the edge, yielding a **55% size reduction** with zero acoustic degradation. However, if deployment constraints necessitate alternative formats—such as uncompressed WAV—the platform mandates a standardized conversion to FLAC during the offline ingestion pipeline, immediately following the file integrity and corruption checks. These optimized FLAC assets are then uploaded to a physically independent storage repository. Depending on the deployment scenario, this backend may correspond to a cloud storage service, an S3-compatible object storage bucket, or any equivalent infrastructure; in our specific implementation, a secure, private server was deployed (see Fig. 3 for illustrative purposes).

- **Decoupled Relational Schema (Data Lineage):** While physical files are stored in the centralized repository, contextual and operational abstraction is managed independently via a PostgreSQL instance (*DB* container in Fig. 9). This architecture is structured explicitly to guarantee absolute data lineage from the target deployment site to the final training split (see Fig. 10). Core hardware and environmental deployment attributes are

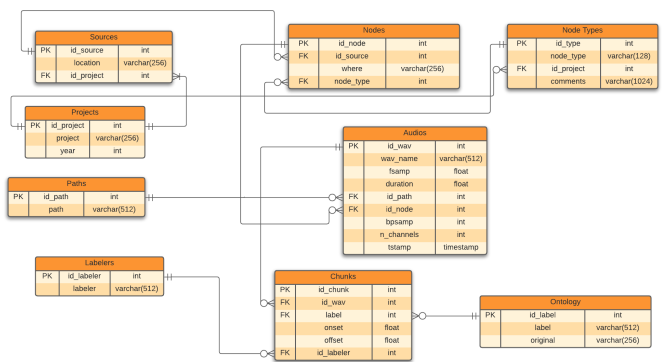


Fig. 10. Overview of the PostgreSQL database schema used to organize the audio data lifecycle in the SOROLL-IA platform. The schema relationally indexes deployment contexts (*Projects*, *Sources*, *Nodes*), file metadata (*Audios*), and human ground-truth segments (*Chunks*, *Ontology*, *Labelers*). Notably, the *id_wav* primary key in the *Audios* table serves as a unique asset identifier irrespective of the underlying audio format, a nomenclature retained for backward compatibility with our baseline prototype. For an in-depth specification of the tables, attributes, and constraints, please refer to our previous work [11].

isolated in the *Projects*, *Sources*, *Node Types*, and *Nodes* tables. To maintain a completely decoupled infrastructure, the metadata of the ingested files is indexed in the master *Audios* registry using file name information for fields *id_node*, *wav_name* and *tstamp* fields, while their concrete physical storage locations across the network are dynamically mapped via the *Paths* table.

- **Infrastructure Resilience and Persistence:** Database administration and backup routines run in isolated containers to prevent contention with the ingestion pipeline. *PGADMIN* provides on-demand management, while *BACKUP* executes scheduled snapshots, ensuring data durability and crash recovery without disrupting active data flows.

C. Structured Metadata and Annotation Management

To address the human labeling bottlenecks and budget constraints defined in Section II-C, the SOROLL-IA metadata layer transitions from a passive storage registry into an active query engine, relying on the relational interplay between the *Audios* and *Chunks* tables (see Fig. 10). While the *Audios* table indexes the global file-level metadata and available acoustic corpus, the *Chunks* table stores the precise temporal events and classes validated by human labelers. Together, these tables form the backbone of the active learning (AL) pipeline: the AL algorithms query their joint state to analyze the current ground-truth distribution and selectively package the most informative, unannotated data segments for human review. The platform operates on an iterative, periodic workflow: operators spend a fixed time window (e.g., one week) annotating the current data batch, and immediately upon its completion, the *INSERTJSON* service commits these new records into the *Chunks* table. This database update automatically triggers the downstream AL selection loop, which processes the newly enriched dataset state to optimize and propose the batch for the subsequent

iteration. This systematic, continuous data preparation loop is driven by two main architectural mechanisms:

- **Relational Metadata Querying:** Operators or algorithms can query segments based on acoustic acquisition parameters, target edge nodes, deployment timelines, or initial model confidence flags, drastically reducing the search space before any annotation batch is generated. This structured filtering directly empowers targeted active sample selection; for instance, in our previous work [11], we adapted the state-of-the-art *Mismatch First-Farthest Traversal* AL algorithm to interface natively with this relational layout. This specific AL approach is highly recommended during the foundational stages of building the ground-truth dataset: the *Farthest Traversal* mechanism guarantees broad acoustic diversity across early data batches, while the *Mismatch First* component provides valuable insights into model performance relative to the currently labeled ground truth. Notably, during the initial iteration where no annotated data is yet available, the algorithm dynamically defaults to exclusive *Farthest Traversal* selection to seed the database.
- **AL Iteration Traceability:** To fulfill strict data lineage requirements, the database schema implements a dedicated tracking extension designed to audit every active AL (see Fig. 11). This centralized tracking layer registers each execution across four highly normalized relational tables:
 - *ALIterations*: Captures the macro-context and boundary conditions of each specific selection run. It records a unique iteration identifier (*id_it*), execution timestamp (*date*), the targeted edge node (*id_node*), and the boundary markers of the processed stream window (*init_wav* and *last_wav*). Furthermore, it logs the ratio of already-labeled audio (*labeled_percentage*), total count of selected files (*audios_proposed*), and links via foreign key (*id_al*) to the specific strategy executed.
 - *AudiosProposed*: Serves as the granular bridge table that maps the chosen assets. It explicitly couples the unique iteration identifier (*id_it*) with the corresponding audio file identifier (*id_wav*) from table Audio in Fig. 11), ensuring clear lineage for every proposed asset.
 - *ALAlgorithms*: Acts as the central algorithm registry, archiving the core metadata of all AL models implemented within the platform. It maps the algorithmic approach through its *type* and *version* attributes, tracking exactly which methodology drove each loop.
 - *ALConfigs*: Manages the polymorphic execution hyper-parameters through a dedicated *json* schema. Linked directly to the iteration via *id_it*, this flexible structure avoids relational rigidity by storing varying parameters unique to each model—such as classifier versions for the *Mismatch* algorithm, or cluster

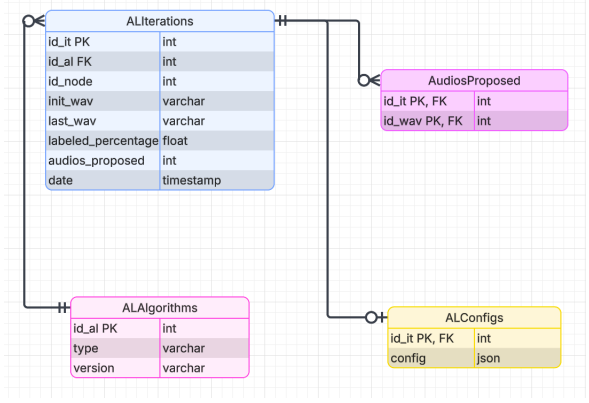


Fig. 11. AL database schema extensions inside SOROLL-IA platform, mapping execution runs (*ALPreprocessing*) to specific selected samples (*AudiosProposed*).

counts and distances for density-based strategies.

D. Human Annotation Infrastructure

To resolve the industrial UX bottlenecks and rigorous pipeline coupling challenges identified in Section II-D, the SOROLL-IA framework introduces a bespoke software solution. While our preliminary work adapted and updated the open-source BAT framework [11], [17], scaling to industrial data volumes required deeper pipeline integration and absolute control over data lineages. Consequently, we developed the New Audio Annotation Tool (NAAT) from scratch (see Fig. 12). NAAT acts as the critical bridge between human expertise and our centralized relational schema, managing the labeling lifecycle through three specialized operational pillars:

- **Role-Based Cross-Validation and Task Management:** The tool ingests the structured batches generated by the active learning layer (*WavsProposed*) and dynamically organizes them into isolated task pools. To guarantee dataset consensus and robust label validation, the infrastructure routes each audio clip to be annotated by an odd number of independent human operators (e.g., a 3-labeler pool), enabling multi-annotator majority consensus. Furthermore, NAAT enforces strict role separation: *Administrators* manage global projects, user credentials, server access protocols, and data export pipelines, while *Annotators* are served targeted audio queues to maximize focus and throughput.
- **Perceptual Audio UX and Multi-Label Taxonomy:** A core design decision of NAAT is the replacement of standard raw waveform views with a high-fidelity, interactive spectrogram-based interface. In acoustic engineering, it is widely established that spectrograms offer superior perceptual grounding compared to raw time-domain signals [18], as visual frequency patterns drastically reduce human cognitive load when isolating overlapping events. This specialized UX enables operators to efficiently perform precise multi-label annotation, drawing dynamic temporal bounding boxes (*onsets* and *offsets*) to isolate

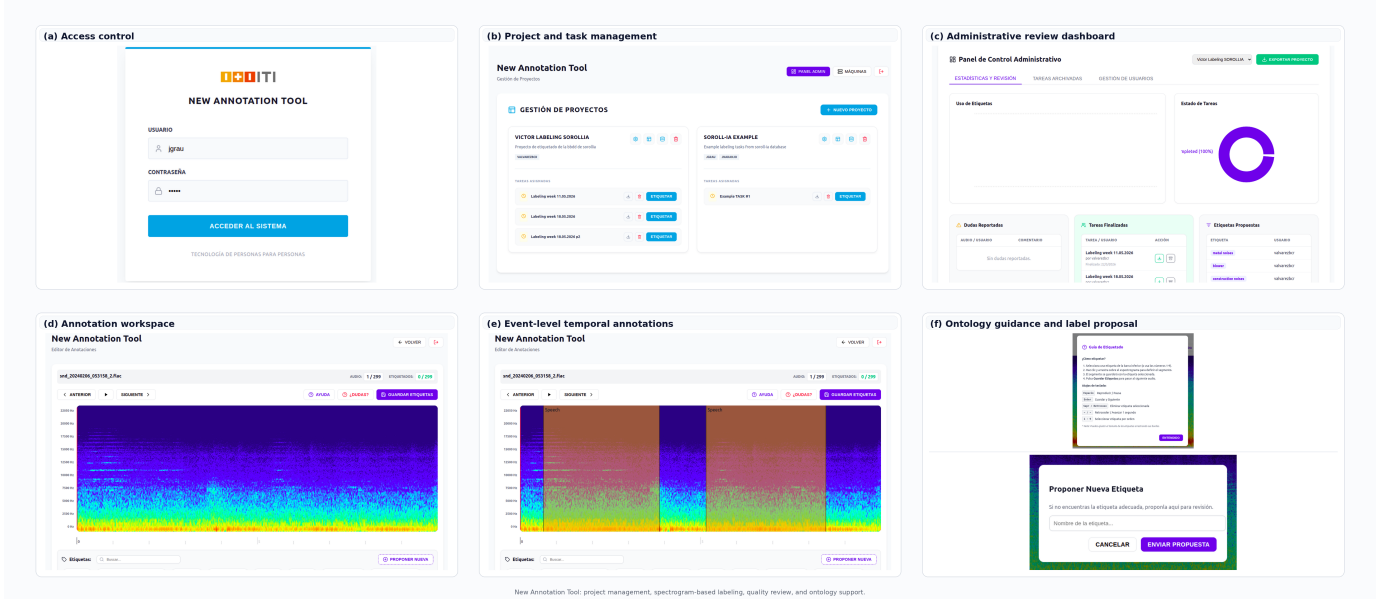


Fig. 12. New Annotation Tool (NAAT) interface components. (a) User authentication, (b) project and task assignment dashboard, (c) administrative oversight with completion metrics and label review, (d) spectrogram-based annotation workspace, (e) event-level temporal segmentation with color-coded labels, and (f) ontology-guided label proposal modal.

simultaneous sound sources. This information could be useful for not only audio tagging but also for sound event detection solutions [19].

- **Ambiguity Protocols, Mitigation, and Closed-Loop Sync:** To prevent the injection of label noise—a critical threat to industrial ML convergence—NAAT implements a strict quality-assurance workflow rather than allowing automated taxonomy updates. When encountering ambiguous signals, annotators can utilize two dedicated mechanisms:

- *Category Proposals:* Annotators can propose new acoustic classes, which are routed to an administrative panel for expert review before being integrated into the master ontology (as seen in Fig. 10 on the table ontology).
- *Doubt Flagging:* Instead of forcing an uncertain label, operators can flag a clip as “doubtful.” This action locks the asset, preventing it from being committed as ground truth. These flagged segments are compiled for weekly alignment meetings between the administration panel and the annotation team [11]. Once the collective ambiguity is resolved, the audio is re-queued and served back to the original labeler for definitive annotation.

Once a batch is finalized and validated, NAAT exports structured JSON packages containing the precise metadata. These files are intercepted by the backend’s *INSERTJSON* service (see Fig. 9), which instantly commits the labels to the PostgreSQL *Chunks* table—binding every temporal event to its original recording file, active ontology class, and operator ID.

E. AI Model Training, Evolution and Versioning

To address the synchronization, reproducibility, and resource constraint challenges outlined in Section II-E, the SOROLL-IA platform incorporates a structured MLOps workflow centered around an integrated MLflow tracking component. The framework couples the periodic dataset expansions with systematic experiment management driven by two technical pillars:

- **MLflow-Driven Experimentation and Run Auditing:** Every iterative training run is registered within MLflow to maintain a clear history of model evolution. For each experiment, the pipeline automatically logs the selected hyperparameters, standard audio tagging metrics ($F1$ -score, mAP), and, crucially, the precise ground-truth dataset snapshot used. Importantly, the unique MLflow Run ID generated during this process is explicitly committed back to the *ALPreprocessing* table described in Section III-C; this explicit relational link guarantees that any future active learning query can be audited back to the exact model state that generated its predictions.

F. Closed-Loop Learning and Continuous Improvement

The architecture sustains long-term model reliability through two core operational workflows:

- **Privacy-Preserving Production Telemetry Ingestion:** Once deployed, edge nodes execute real-time local inference while continuously piping lightweight metadata back to the central backend. To protect production bandwidth and comply with the strict privacy regulations detailed in Section II-A, raw audio is not streamed continuously. Instead, the edge nodes send asynchronous, low-overhead payload logs containing timestamps, node

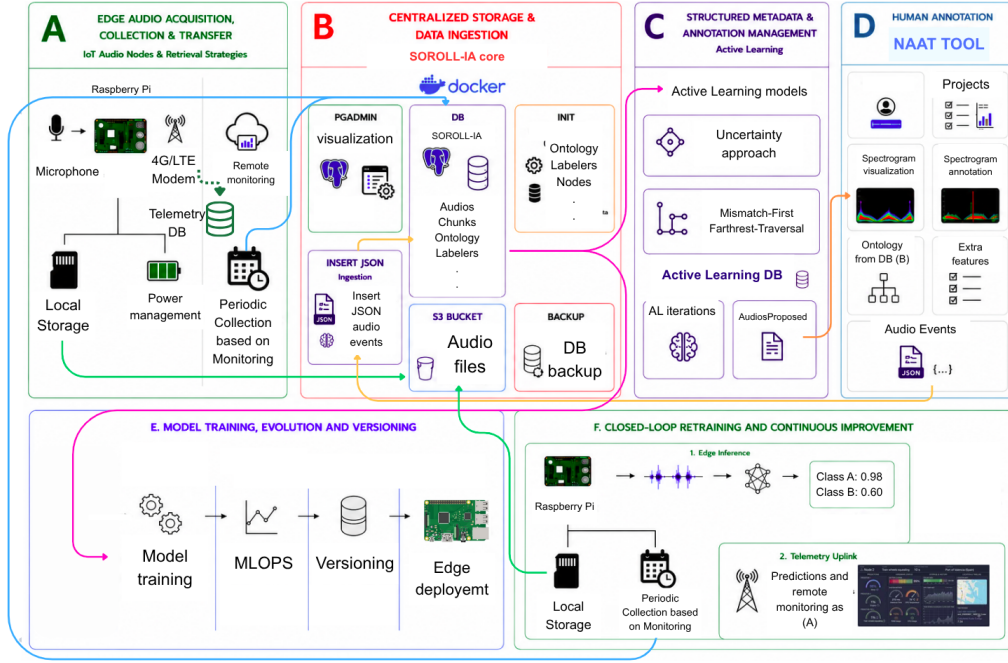


Fig. 13. Complete end-to-end SOROLL-IA audio AI pipeline architecture. The diagram illustrates the unified interplay between edge acquisition (Block A), central ingestion (Block B), active learning metadata orchestration (Block C), human annotation (Block D), and MLOps evolutionary deployment (Blocks E and F). Green lines signify physical raw audio and telemetry data transfers; blue lines denote the ingestion of file metadata into the central SOROLL-IA database; and magenta lines represent the relational queries dispatched from the *Chunks* and *Audios* tables to orchestrate active learning iterations and trigger downstream model retraining cycles.

identifiers, predicted acoustic classes, execution latencies, and model confidence scores (see Fig. 8).

- **Data Drift Detection and active learning Curation:**

The central ingestion/monitoring pipeline stores these incoming confidence scores to detect anomalies, high-uncertainty inferences, or symptoms of domain drift. In long-term deployments, it is highly recommended to dynamically shift the AL strategy from a diversity-driven approach to a purely model-centric, uncertainty-based paradigm [20]. For instance, when a node has operated in the field for a considerable timeframe (e.g., more than 6 months) without retraining, the underlying acoustic environment may evolve due to seasonal shifts or mechanical wear, triggering domain drift. Because the node’s real-time edge inferences and confidence logs are already systematically archived in the central database, the pipeline can execute uncertainty-based AL queries over these fresh, uncollected periods. Consequently, when field technicians proceed to physically retrieve the local SD card, the system can instantly propose a highly targeted validation batch composed exclusively of assets that exhibited the highest predictive entropy in production, maximizing the labeling budget exactly where the model’s confidence is weakest.

Once domain experts resolve ambiguities and validate these critical production samples through NAAT’s cross-validation protocol, the resulting high-fidelity labels are committed back into the PostgreSQL registry (see Fig. 9 DB container). This

action systematically closes the operational loop: the database couples these new assets with the historical lineage, automatically re-triggering the MLflow pipeline to train an updated, hardened model version. This new model run is assigned a fresh identifier that is linked back to the next AL cycle, while its optimized topology is automatically compiled into ONNX format for subsequent edge deployment. Rather than executing uncontrolled, computationally expensive retraining cycles, this closed-loop structure provides a highly governed engineering standard for continuous improvement, ensuring full traceability and robust data-model synchronization across the entire industrial audio AI lifecycle.

IV. DISCUSSION AND CONCLUSIONS

This paper introduces the SOROLL-IA Platform, a unified, end-to-end operational framework that transitions industrial audio tagging from a collection of isolated machine learning tasks into a cohesive, continuous data-and-model lifecycle (see Fig. 13). While existing literature frequently isolates acoustic modeling from data engineering, this work systematically bridges the gap between hardware constraints and software orchestration. By analyzing the architecture, some core engineering insights emerge, demonstrating how the platform addresses the lifecycle challenges from initial data ingestion to edge deployment:

- **Decoupled Infrastructure and Active Orchestration:**

Managing massive industrial databases of unstructured audio assets is unfeasible without separating physical

storage from contextual metadata. SOROLL-IA proves that a dedicated relational layer does not merely store logs; it acts as an active gateway. By structuring a weekly pipeline lifecycle, the database orchestrates downstream active learning (AL) algorithms—such as our adapted *Mismatch First-Farthest Traversal*—to optimize the search space and guarantee data diversity from the very first annotation batches.

- **Quality-Assured Human Infrastructure:** Human annotation is the primary bottleneck in large-scale ML deployment. Through the custom-built New Audio Annotation Tool (NAAT), the platform shifts away from raw waveform visualization toward perceptually grounded, interactive spectrograms that drastically reduce cognitive load. More importantly, the implementation of an odd-number cross-validation pool, coupled with strict “doubt-flagging” protocols and weekly alignment meetings, prevents the injection of label noise into the master PostgreSQL registry, ensuring that only high-fidelity ground truth is committed.
- **Traceable MLOps and Drift Mitigation:** Deploying a model is not the endpoint, but the activation of a continuous improvement loop. By integrating an MLflow tracking component, every training run is mathematically audited, linking specific multi-label dataset snapshots and cross-validation splits directly to a unique Run ID, which is committed back to the AL tracking tables. Once compiled into optimized ONNX artifacts for edge deployment, these models log real-time confidence scores. This allows the centralized pipeline to monitor and detect domain drift (e.g., after 6 months in production), automatically shifting the AL strategy toward an uncertainty-based paradigm that isolates high-entropy clips for targeted physical recovery and retraining.

In conclusion, the primary contribution of this work is the structural validation of a complete operational cycle tailored to the modern demands of large-scale audio database management. By mapping explicit hardware constraints to microservice-driven backend solutions, SOROLL-IA successfully transitions raw environmental noise into an organized, traceable, and self-enriching AI asset. The robustness of this ecosystem is demonstrated through a continuous lifecycle that spans from edge acoustic recording and storage-optimized decoupled ingestion, to relational active learning querying, human annotation packaging, model retraining, and subsequent edge redeployment. Ultimately, this work delivers a concrete, production-ready blueprint that directly addresses the critical challenges in the management and delivery of large audio databases—proving that the bottleneck of massive acoustic data ingestion can be systematically resolved through decoupled infrastructure, strict relational lineage, and model-guided data curation.

REFERENCES

- [1] F. Schmid, K. Koutini, and G. Widmer, “Efficient large-scale audio tagging via transformer-to-cnn knowledge distillation,” in *ICASSP 2023-2023 IEEE international Conference on acoustics, Speech and signal processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [2] S. Singh, C. J. Steinmetz, E. Benetos, H. Phan, and D. Stowell, “Atgnn: Audio tagging graph neural network,” *IEEE Signal Processing Letters*, vol. 31, pp. 825–829, 2024.
- [3] K. Zaman, M. Sah, C. Direkoglu, and M. Unoki, “A survey of audio classification using deep learning,” *IEEE access*, vol. 11, pp. 106 620–106 649, 2023.
- [4] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.
- [5] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, “FSD50K: An open dataset of human-labeled sound events,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2022.
- [6] K. J. Piczak, “Environmental sound classification with convolutional neural networks,” in *Proceedings of the IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2015, pp. 1–6.
- [7] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “Panns: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.
- [8] A. Singh, H. Liu, and M. D. Plumbley, “E-panns: sound recognition using efficient pre-trained audio neural networks,” in *Inter-Noise and Noise-Con Congress and Conference Proceedings*, vol. 268, no. 1. Institute of Noise Control Engineering, 2023, pp. 7220–7228.
- [9] M. Garcia-Ballesteros, J. Naranjo-Alcazar, J. Grau-Haro, R. Ribes-Serrano, and P. Zuccarello, “Confidently Wrong: Evaluating AudioSet-Trained Models Under Real-World Deployment,” in *Journal of the Audio Engineering Society*, no. 10274. AES, jun 2026.
- [10] R. Tanabe, H. Purohit, K. Dohi, T. Endo, Y. Nikaido, T. Nakamura, and Y. Kawaguchi, “Mimii due: Sound dataset for malfunctioning industrial machine investigation and inspection with domain shifts due to changes in operational and environmental conditions,” in *2021 IEEE workshop on applications of signal processing to audio and acoustics (WASPAA)*. IEEE, 2021, pp. 21–25.
- [11] J. Naranjo-Alcazar, J. Grau-Haro, R. Ribes-Serrano, and P. Zuccarello, “A data-centric framework for machine listening projects: Addressing large-scale data acquisition and labeling through active learning,” in *Future of Information and Communication Conference*. Springer, 2025, pp. 647–659.
- [12] Y. Sun, Q. Xu, Y. Su, Q. Zhu, Y. Dou, X. Liu, and K. Xu, “Audioset-r: A refined audioset with multi-stage llm label reannotation,” in *Proceedings of the 33rd ACM International Conference on Multimedia*, ser. MM ’25. ACM, Oct. 2025, p. 13089–13096. [Online]. Available: <http://dx.doi.org/10.1145/3746027.3758260>
- [13] C.-C. Huang, P.-H. Chen, C.-I. Hsu, H.-y. Lee, D. Yu, and H.-M. Lee, “On the effectiveness of domain adaptation for environmental sound classification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7269–7273.
- [14] Pi Supply, “Pijuce hat,” <https://github.com/PiSupply/PiJuice/>, 2025, accessed: 2026-06-26.
- [15] RØDE Microphones, “Lavalier ii,” <https://rode.com/es/products/lavalier-ii>, 2025, accessed: 2026-06-26.
- [16] BROVI, “E3372-325 usb cellular modem,” <https://www.brovi-tech.com/productshow.php?cid=2&id=248>, 2025, accessed: 2026-06-26.
- [17] B. Melendez-Catalan, “BAT: Basic audio tool,” <https://github.com/BlaiMelendezCatalan/BAT>, accessed: 2026-06-09.
- [18] M. Cartwright, A. Seals, J. Salamon, A. Williams, S. Mikloska, D. MacConnell, E. Law, J. P. Bello, and O. Nov, “Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 1, no. CSCW, pp. 1–21, 2017.
- [19] A. Mesaros, T. Heittola, T. Virtanen, and M. D. Plumbley, “Sound event detection: A tutorial,” *IEEE Signal Processing Magazine*, vol. 38, no. 5, pp. 67–83, 2021.
- [20] Y. Wang, A. E. M. Mendez, M. Cartwright, and J. P. Bello, “Active learning for efficient audio annotation and classification with a large amount of unlabeled data,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 880–884.