AudioInsight: Online exploration of Large Audio Datasets for Musical Acoustics

Michael Starakis
Department of Music Technology and Acoustics
Hellenic Mediterranean University
Rethymnon, Greece
ddk216@hmu.gr

Chrisoula Alexandraki
Department of Music Technology and Acoustics
Hellenic Mediterranean University
Rethymnon, Greece
chrisoula@hmu.gr

***Abstract*—New perspectives in correlating physical and perceptual characteristics of musical instruments have emerged through recent advancements in machine learning and high-performance computing. Driving such correlating processes, for instance through training deep learning models, requires the assembly of large audio datasets and the development of tools for Exploratory Data Analysis (EDA). This article presents AudioInsight, a web application for the audio-visual exploration of sound datasets for musical acoustics. This application is a by-product of our research on drumhead acoustics, which aims at the computational inference of damping strategies resulting in a desired sound texture. Developed in Python and Dash-Plotly, AudioInsight features an interactive and responsive user interface providing tools for dataset exploration, statistical analysis and visualization of parameter distributions and relevant correlations. Moreover, AudioInsight permits generating dataset clusters and exploring sounds by interactively navigating within their 2D or 3D graphic representations. Future enhancements of this application, allowing musicians and researchers to contribute to musical instrument datasets through crowdsourcing, may serve as an example application within the Internet of Sounds (IoS) ecosystem.**

Keywords—musical acoustics, drumhead acoustics, Exploratory Data Analysis (EDA), audio datasets, audio clustering, dataset visualization, interactive visualization

I. INTRODUCTION

State-of-the-art AI-driven techniques have enhanced the ability to process and interpret large datasets of acoustic signals, providing new insights into the physical and perceptual aspects of musical instruments. Such methodologies not only broaden the scope of research in musical acoustics but also foster the development of highly novel applications involving collaborative exchanges across different research domains.

Research endeavors in this direction have highlighted the need for large and high-quality sound datasets. These datasets are essential for applications such as estimating physical characteristics from sound and training advanced deep learning models for sound source recognition. Ensuring the validity of these datasets, not only in terms of sound quality but maybe more importantly in terms of their descriptive annotations associating physical and perceptual traits, requires the development of tools for Exploratory Data Analysis (EDA). A significant challenge in this direction relates to finding appropriately annotated audio segments, which are crucial for training effective machine learning models for the task at hand. While large collections of audio data are currently available, finding datasets with accurate and comprehensive annotations, linking physical properties to perceptual characteristics is highly cumbersome. This challenge is intensified by the abundance of data and the nuanced nature of sound annotations, which often require human curation and expert knowledge to be accurate and useful. Therefore, tools assisting researchers to process and analyze large datasets, as well as to ensure the quality and relevance of accompanying annotations, are of fundamental importance.

The concept of the Internet of Sounds (IoS) emerges as a relevant framework in this context [1]. Besides environmental acoustics, sensor networks, and smart musical instruments, IoS can contribute to musical instruments studies by integrating diverse sound databases, making it possible to access and utilize a wide range of sound samples and related data from different sources, including musicians cell phones. This interconnected framework can support more comprehensive and multifaceted studies in musical acoustics, providing researchers with a richer set of data and tools to work with.

The AudioInsight application presented in this article was developed to address our research requirements with respect to the exploration of sound datasets for drumhead acoustics. At present, the application leverages a dataset of over 11,000 synthetic drumhead sounds generated through Finite Difference Time Domain (FDTD) synthesis [2]. The tools provided by the AudioInsight application allow visualizing and analyzing sound data, facilitating a deeper understanding and helping researchers and musicians to identify interesting sounds, patterns and correlations that might be overlooked by conventional analysis methods.

The rest of this article is structured as follows: Section II discusses similar applications for EDA in different research domains, highlighting their strengths and limitations. Then, section III presents our motivation driving the development of the AudioInsight application for drumhead acoustics. It discusses the requirements of analyzing large-scale audio data and the limitations of conventional methods to do so. Section IV elaborates on the design and the functionality of the application and section V provides implementation details. Finally, the article discusses future perspectives and summarizes key contributions of our research.

II. RELATED WORK

A. EDA for large datasets

EDA is a fundamental methodology in data science, introduced by Tukey [3], designed for summarizing, visualizing, and understanding complex datasets. The primary objective of EDA is to uncover underlying patterns, spot anomalies, test hypotheses, and check assumptions through graphical representations and robust statistical approaches in order to understand data [4].

EDA is widely recognized for its critical role in various fields, including musical acoustics. Techniques such as data visualization, including histograms and scatter plots, statistical summaries and dimensionality reduction are commonly employed in EDA to gain insight to large, multidimensional audio datasets. Large datasets often contain complex structures and relationships that can be difficult to understand without exploratory techniques. By summarizing key aspects of the data visually and statistically, EDA helps analysts to efficiently uncover trends and correlations that otherwise might be missed [5], [6], [7].

B. *EDA Tools and Applications*

Numerous tools have been developed to support EDA across various fields, each addressing the unique needs of different types of data and analysis workflows. Specialized platforms have been developed to handle domain-specific data in the fields of environmental research [8], weather forecasting [9], seismic research [10], economics [11], and medical research [12], [13].

In data science, the rise of computational notebooks such as JupyterLab [1] and RStudio [2] has revolutionized how researchers conduct EDA [14]. By integrating code, documentation, and outputs within a single environment, library packages such as Matplotlib, Plotly, Seaborn, Tidyverse, Pandas, and Scikit-learn offer robust capabilities for data manipulation, visualization, and machine learning experimentation. Similarly, EDAssistant [15] integrates computational notebooks to provide in-situ code search and recommendation, thus enhancing the capabilities offered for data exploration. In the domain of big data, platforms like Apache Spark and Hadoop [16] provide distributed computing solutions for processing vast datasets. However, these frameworks require programming skills and specialized knowledge to explore any type of data.

Higher level visualization-centric tools such as Tableau [17], and PowerBI[3] are favored for their intuitive interfaces and powerful data visualization capabilities, facilitating complex charts and dashboards interactively. Additionally, frameworks like Streamlit[4], Dash[5], and Panel[6] enable the development of interactive web-based applications for EDA. In terms of programmatic approaches, Vega-Lite [18] offers a high-level grammar for creating interactive graphics, while the Voyager browser [19], provides a gallery of automatically generated visualizations. Furthermore, recent EDA tools such as Pandas Profiling[7], generate interactive HTML reports to summarize dataset statistics and visualizations.

When it comes to audio datasets, their data exploration and analysis presents unique challenges due to high dimensionality and temporal substance of audio data. There are several tools for analyzing and annotating single audio files, such as standalone software (e.g. Audacity, Sonic Visualiser [20], Praat) and Python libraries (e.g. Librosa [21], Essentia [22]).

Web-based applications like D-Tale [8] provide an interactive interface for visualizing and exploring pandas dataframes. The TensorFlow Projector [9] and Dash-based applications, like t-SNE Explorer[10] and Clustergram[11] stand out as powerful platforms for visualizing high-dimensional data, facilitating the discovery of patterns and clusters. Additionally, the Speech Data Explorer developed by NVIDIA [23], also a Dash-based application, offers interactive exploration of automatic speech recognition and text-to-speech datasets, focusing on global statistics and error analysis. Finally, Google Creative Lab offers web-based interactive experiments for exploring audio in the context of different domains. For example, projects like Drum-Machine[12] leverage the t-SNE algorithm for sound clustering. Spectrogram-and-Oscillator [13] and Bird-Sounds [14] provide interactive experiences through FFT spectrum analysis and audio-visual integration.

Data-driven approaches like Data2Vis [24], DeepEye [25] and VizML [26] automatically generate and recommend visualizations based on deep learning techniques. Similarly, ATENA [27], a system that auto-generates EDA notebooks using deep reinforcement learning, demonstrates the potential for automation in EDA processes. Researchers like Heise and Bear have employed unsupervised methods to reveal natural clustering patterns in audio data [28]. Lastly, Fallgren [29] presented tools supporting nonsequential browsing and data interpretation for exploring large amounts of audio data.

Despite the continuous evolution of EDA tools, there are no comprehensive platforms specifically designed for the exploration of large-scale audio datasets. Existing applications often focus either on general analysis for structured datasets or on specific audio processing tasks of individual audio files. They rarely provide a user interface suitable for the multifaceted nature of audio data, without requiring significant coding expertise. This situation underscores the importance of developing applications like AudioInsight, which aims to provide a unified platform for EDA of audio datasets.

III. MOTIVATION: DRUMHEAD ACOUSTICS

The development of the AudioInsight application emerged from our research on drumhead acoustics [30], [31]. The objective of this study was to computationally infer the damping material that needs to be applied on the surface of a membrane to achieve a given sound texture. This inverse acoustic problem presented unique challenges, particularly in handling and exploring vast amounts of data. AudioInsight was conceived as a solution to these challenges, to provide a versatile visual and auditory exploration tool, capable of depicting complex relationships between physical properties and sound characteristics of drumheads. In the following, we briefly review this study to explicitly outline the challenges encountered during our research and to showcase an example demonstrating the capabilities of the application.

---

[1] https://jupyter.org/
[2] https://posit.co/products/open-source/rstudio/
[3] https://www.microsoft.com/en-us/power-platform/products/power-bi/
[4] https://streamlit.io/
[5] https://dash.plotly.com/
[6] https://panel.holoviz.org/
[7] https://pandas-profiling.github.io/pandas-profiling/
[8] https://github.com/man-group/dtale/
[9] https://projector.tensorflow.org/
[10] https://dash.gallery/dash-tsne/
[11] https://dash.gallery/dash-clustergram/
[12] https://experiments.withgoogle.com/drum-machine/
[13] https://experiments.withgoogle.com/chrome/spectrogram-and-oscillator/
[14] https://experiments.withgoogle.com/ai/bird-sounds/view/

A. Methodology

The drumhead dataset comprises 11,114 synthetic drumhead sounds, which were generated using a FDTD algorithm modelling the behavior of a vibrating circular membrane. The model assumes the distribution of malleable, paste-like material on the surface of the membrane, which alters its vibrational behavior, its modal frequencies, and thus the sound it generates. Paste distribution was varied according to six patterns that were inspired by handcrafted and commercial drum dampeners that are commonly used by percussionists. These patterns are shown on Fig. 1. Different sounds were generated by varying the area covered by paste in each pattern (e.g. by varying the width of the lines), the amount of paste mass per unit area as well as the strike position of membrane excitation. Three points were chosen as strike positions: one near the center, one near the perimeter and one in between. The wide range of parameters and their variations resulted in a set of sounds that reflects the complex interactions between damping materials, membrane properties, and excitation positions, making it suitable for exploratory analysis and machine learning tasks.

To explore whether similar damping schemes would result in similar sounds, several clustering methods (PCA, t-SNE, LDA, and more) were employed. Clustering methods allow visualizing high-dimensional data in lower-dimensional spaces (i.e. 2D, 3D), thus permitting the identification of distinct sound clusters grouping similar sounds together and providing insight into the influence of paste patterns and impact points on the resulting sound signals. For example, Fig. 2, presents two 2D projections of the audio dataset. Each point represents a different sound signal and each color depicts a different damping pattern. Here the axes X, Y represent to one of the possible 2D planes of the multidimensional data, chosen by the clustering algorithm to more effectively depict groups of similarities among the sound signals.

In our original research, data exploration was used to guide our efforts towards an inverse acoustic problem, namely to infer the damping strategy corresponding to a desired sound. This problem was approached by training a two-output Convolutional Neural Network (CNN) performing pattern identification through a classification branch and estimation of the amount of added paste through a regression branch. The CNN model demonstrated remarkable accuracy in identifying damping configurations from sound input [30], thus confirming the capability of data-driven approaches to address problems of musical acoustics.

B. Data Exploration Challenges

As our research progressed, we encountered significant challenges in exploring and analyzing the dataset. As the dataset grew to over 11,000 samples, the complexity of the analysis increased dramatically. Navigating through thousands of audio files can be overwhelming and time-consuming, making it difficult to gain a systematic overview of the data at hand. Conventional methods of audio analysis that work well for small samples, become impractical, inefficient, and prone to human error when applied to datasets of this size. While comparing two sounds might be straightforward, scaling this comparison to thousands of sounds can lead to fatigue, making it easy to overlook critical patterns and relationships.

This challenge was intensified by the fact that each sound sample was associated with numerous parameters defining physical properties, i.e. damping variables for paste patterns, strike points and relevant perceptual features, e.g. frequency content, amplitude envelopes, etc. When dealing with such high parameter spaces it is very likely that certain value ranges are under-represented for parameters that need to be accurately estimated through regression. Moreover, subtle patterns and correlations between different parameters and sound characteristics are hard to identify. Conventional 2D plots and charts are often insufficient to capture these complex

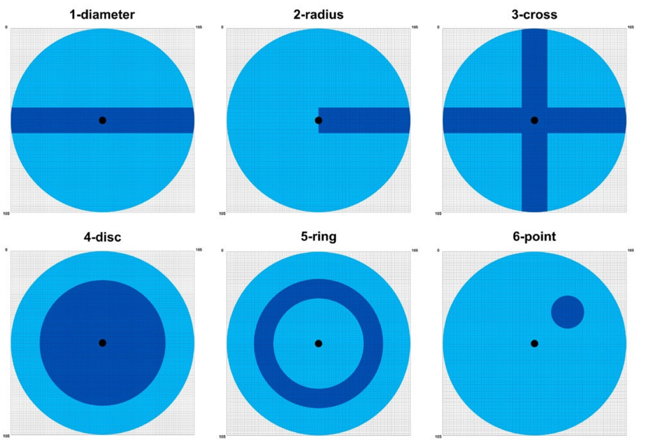

Fig. 1. The paste patterns used for generating the dataset of drumhead sounds.

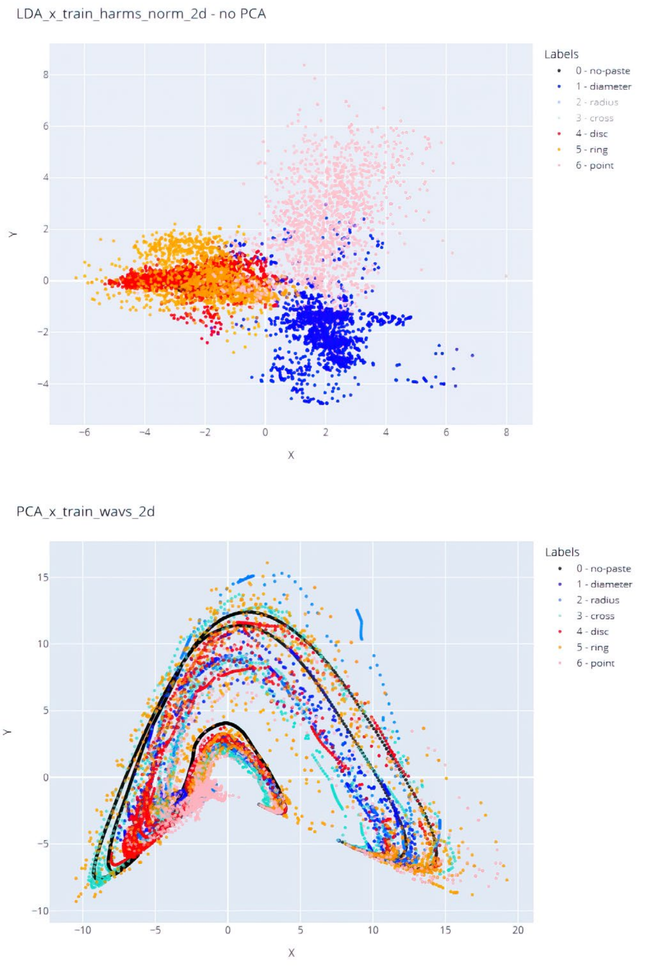

Fig. 2. 2D clustering plots obtained by applying LDA (top) and PCA (bottom) algorithms. Different colors correspond to different paste pattern cases. Each point on the plot represents a different sound from the 11,114 synthetic drumhead sounds.

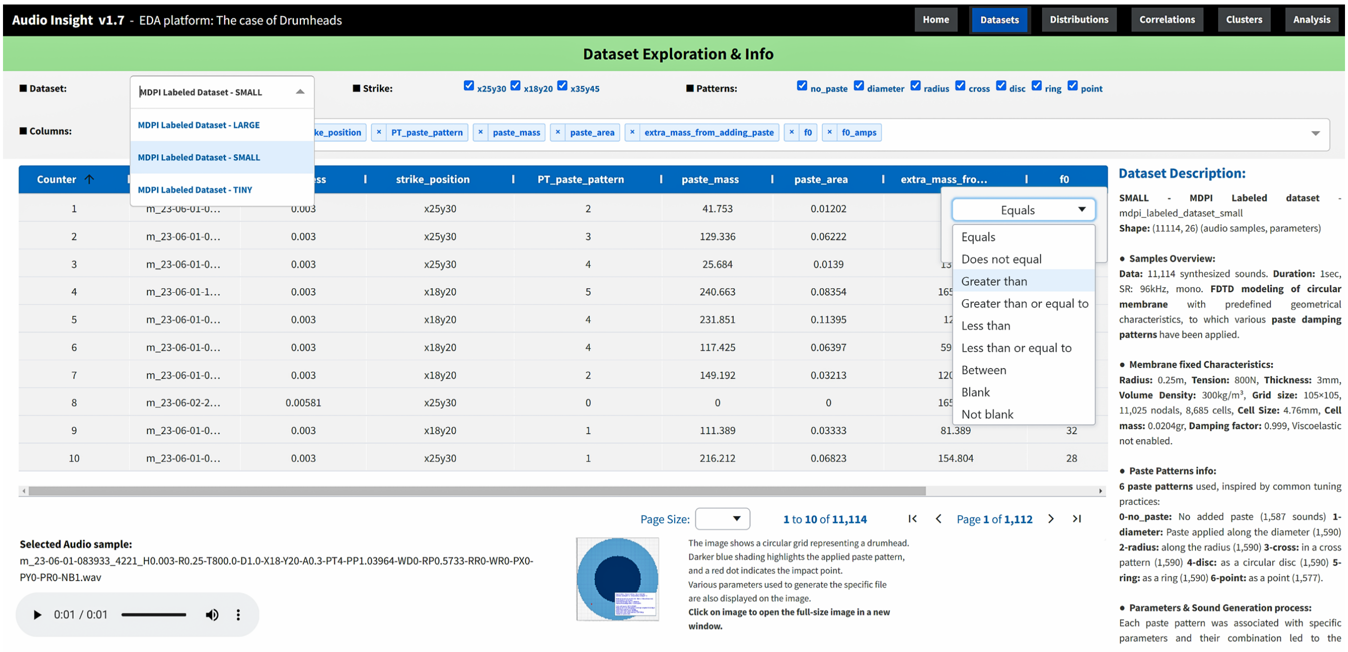

Fig. 3. Dataset page. The interactive table of the dataset contents, with customization options.

non-linear relationships. The effect of paste patterns on sound characteristics, for example, was not easy to deduce, making it challenging to predict or understand the acoustic results of specific damping configurations without thorough analysis.

To address these challenges, AudioInsight was developed as a demo application focusing on drumheads. Through its interactive visualizations and clustering capabilities, the application allowed us to uncover complex, non-linear relationships between physical properties and perceptual features that might otherwise remain hidden. These insights not only advanced our research objectives but also demonstrated the potential of AudioInsight as a powerful tool for systematic EDA of large audio datasets in musical acoustics.

IV. The AudioInsight Application

The AudioInsight application is available online[15]. Its design goals center around providing a powerful, yet easy-to-use, interactive system for thorough data exploration and analysis, accessible to researchers, instrument manufacturers and musicians. The visual and audio analysis, along with tools for clustering representations, audio previews, and feature graphs, help users uncover patterns and correlations that are not obvious through regular methods and can be particularly valuable for machine learning tasks.

For instrument manufacturers and musicians, AudioInsight may guide the instrument design and tuning process, while providing a tool to better understand the acoustic properties of their instruments. It allows musicians and luthiers to search for similar sounds, explore relationships between their timbral discrepancies and to systematically enhance their tacit knowledge.

The user interface of the application is organized in six pages. Except from a home page that highlights key features of our research, the other pages present the dataset according to selected descriptive parameters. The pages offer dropdown menus for users to select their desired dataset and choose specific columns (parameters) for analysis. For categorical parameters, which in the case of membranes are the paste patterns (Fig. 3) and the strike positions, the graphical representations may be filtered to display data distributions of specific labels. This functionality is provided through checkboxes. For continuous variables, e.g. mass of added paste, fundamental frequency (f0) of the sounds, etc., scaling, zooming and panning of the graphs may be used to focus on certain value ranges.

The functionality of these pages is briefly described in the following subsections.

A. Dataset Page

The Datasets page (Fig. 3) presents an interactive table of the dataset contents, with options to filter and customize the display. Rows correspond to dataset instances, i.e. sound files and columns to descriptive parameters, in this case physical parameters describing the membrane and the damping material, the strike position on the membrane, etc. The table uses data pagination and allows users to show/hide specific columns, sort, filter, search for specific parameter values and more.

Clicking on a row triggers audio playback of the corresponding sound file and displays its corresponding drumhead grid image, which visually represents the paste pattern applied to the drumhead, appearing below the table on Fig. 3. The right-side panel presents a description of the dataset and provides detailed information about the data being explored.

B. Distributions Page

The Distributions page (Fig. 4) helps users visualize and analyze the statistical distributions of various parameters within the sound dataset. The page allows selecting some parameter and displaying its distribution either across the entire dataset or by filtering out certain labels (categorical

[15] http://musicolab.hmu.gr:8050

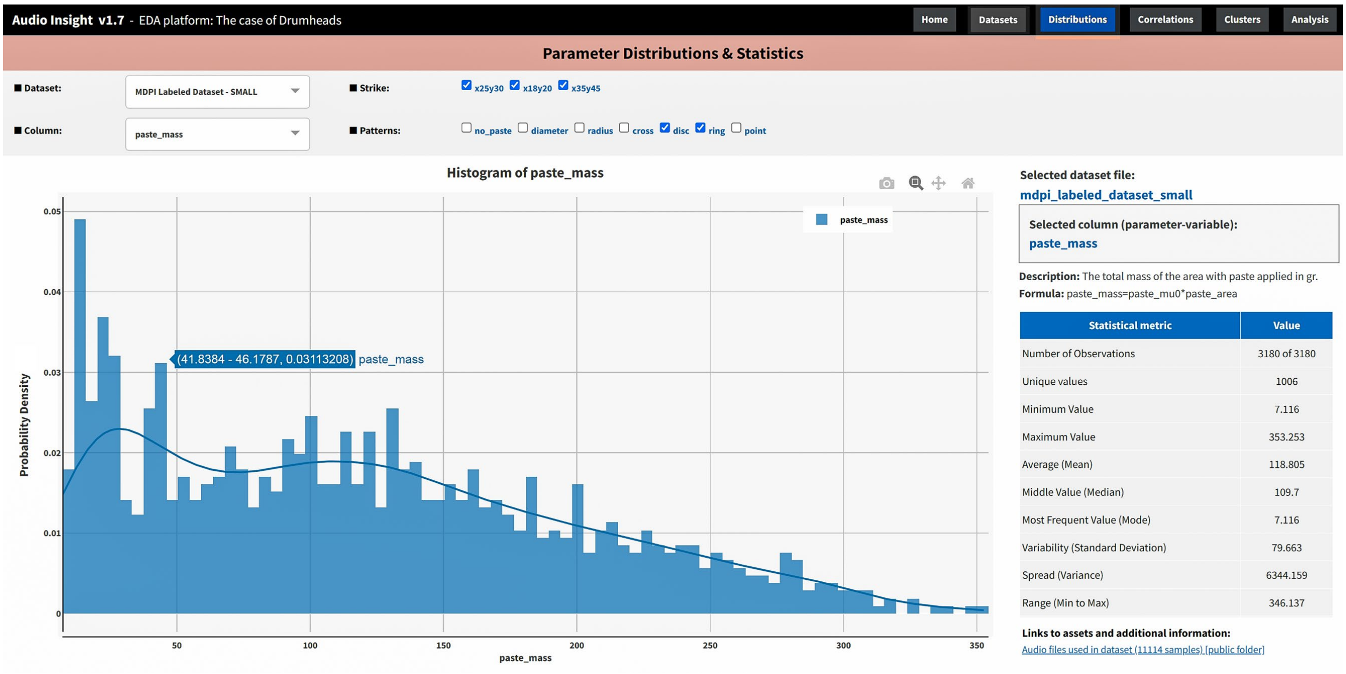

Fig. 4. Distributions Page. The distribution (probability density) of paste mass for the circular patterns ring and disc and a statistical summary appearing on the right.

parameters). For example, the graph on Fig. 4 displays the distribution (probability density) of paste mass, when paste is distributed using circular patterns i.e., ring and disc. Hovering over the plot displays the y-axis value and the range of x-axis values represented by each bar of the histogram. Users can customize the appearance of the plots through scaling, zooming, and panning, and they can also save the plot as an image to disk.

The right-side panel provides context about the parameter being visualized, including its calculation formula and statistical metrics like minimum, maximum, median, standard deviation, etc. This statistical summary allows researchers to identify patterns, trends, and assess whether the dataset is balanced with respect to different parameters, hence allowing to weigh the suitability of data for the task under investigation.

C. Correlations Page

The Correlations page (Fig. 5) provides tools for exploring and visualizing relationships between pairs of parameters in the sound dataset. It features an interactive scatter plot, complemented by a range of customization options. Users are presented with dropdown menus allowing to select the parameters represented on x-axis, and y-axis, along with checkboxes for filtering out certain values of categorical data.

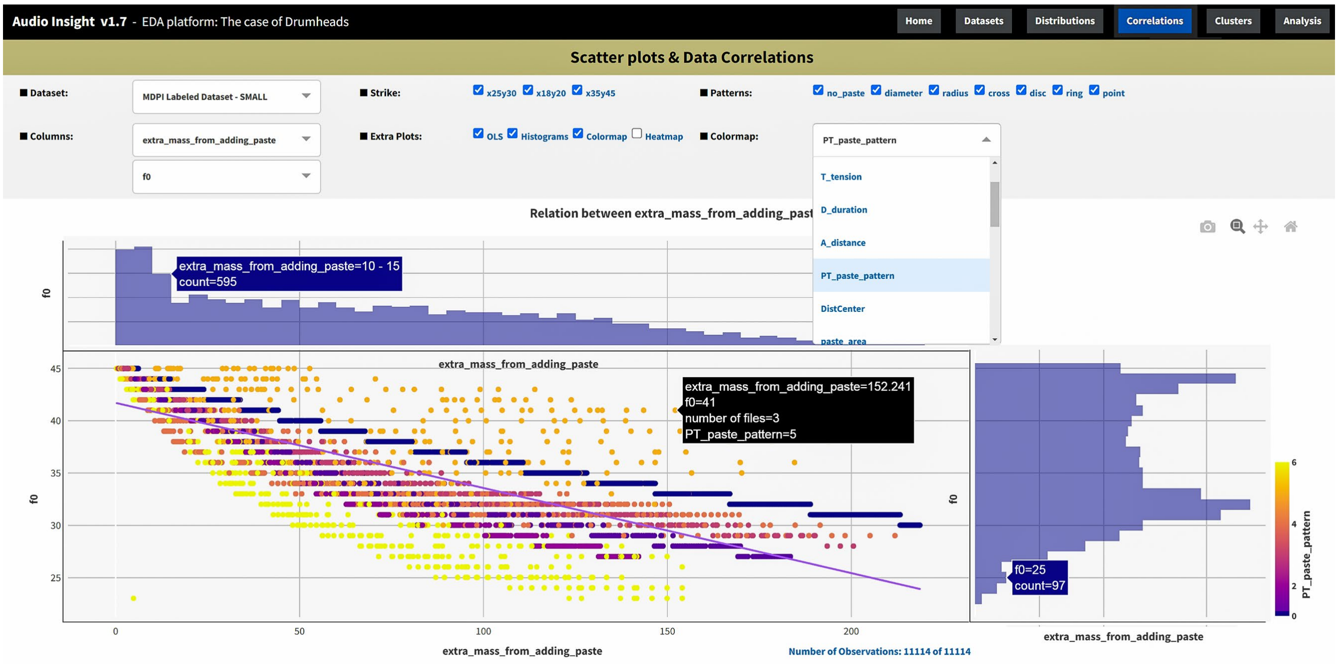

Fig. 5. Correlations page. The correlation plot between the extra mass from paste application and f0 of the sounds. Each color from the colormap represents a different paste pattern case. Histograms of the both variables are also shown, with mouse hover information corresponding to each point on the plots.

The plot updates in real-time, allowing for instantaneous visualization of correlations. Additional features include a Kernel Density Estimation (KDE) line, histograms along the axes, and colormaps based on a third parameter, enhancing the depth of analysis. Users can also toggle a heatmap view for density visualization.

For example, Figure 5 illustrates a correlation plot showing how the additional mass applied via paste relates to the fundamental frequency (f0) of the resulting sound. Each paste pattern case is distinguished by a different color on the plot. Additionally, histograms depicting the distributions of both variables are provided, above and to the right of the scatter plot.

This diagram allows making various hypotheses on how increasing the mass of the membrane alters its perceived pitch. First, it makes apparent that small variations of the amount of added paste in the same pattern does not change the value of f0. This may either indicate some precision error of the FDTD algorithm used to generate the sounds or suggest a hypothesis that needs to be physically and perceptually tested.

Furthermore, and according to this diagram, different patterns result in different f0/mass correlations. For most patterns, besides (point) yellow and orange (ring), increasing membrane mass via paste will linearly (i.e. by the same amount) decrease f0, and hence the observed pitch. For paste patterns point and ring there is no such relation, as the same mass may result in different f0 values, indicating that the f0 may be determined either by the strike position or the surface area covered with paste or that there is some error in parameter assignment or sound generation. There are numerous additional hypotheses that may be driven by the graphs of Fig. 5, which however are beyond the scope of this article.

Again, the plot supports zoom, pan, and hover functions to display detailed information about individual data points, such as values of the x and y axes, the number of files corresponding to each point on the plot, and paste pattern cases. Finally, there is a counter at the bottom of the page that indicates the number of observations (i.e. sound files) being displayed.

D. Clusters Page

The Clusters page (Fig. 6) provides a tool for audiovisual navigation within the dataset by using advanced clustering techniques. It features a central plot showing the data points resulting from the applied clustering method on the input data set. Literally applied on the raw values of wav audio data. The clustering plot can be viewed in 2D or 3D rendering. The following clustering methods are currently supported: Principal Component Analysis (PCA) [32], t-Distributed Stochastic Neighbor Embedding (t-SNE) [33], [34], Pairwise Controlled Manifold Approximation (PaCMAP) [35], Uniform Manifold Approximation and Projection (UMAP) [36] and Linear Discriminant Analysis (LDA) [37]. Users can specify the number of files to be previewed and filtered according to categorical data.

The plot is fully interactive, allowing users to zoom, rotate, and pan, as well as save a view of the plot as an image to disk. Hovering over data points on the plot triggers real-time audio playback of the corresponding sound file, providing an auditory dimension to the visual exploration. A right-side panel displays details about the parameters of each sound file. Information concerning the utilized clustering method, as well as details about the input dataset are additionally provided.

The Clusters Page is designed to combine visual, auditory, and textual information, thus suggesting a multisensory approach to data exploration.

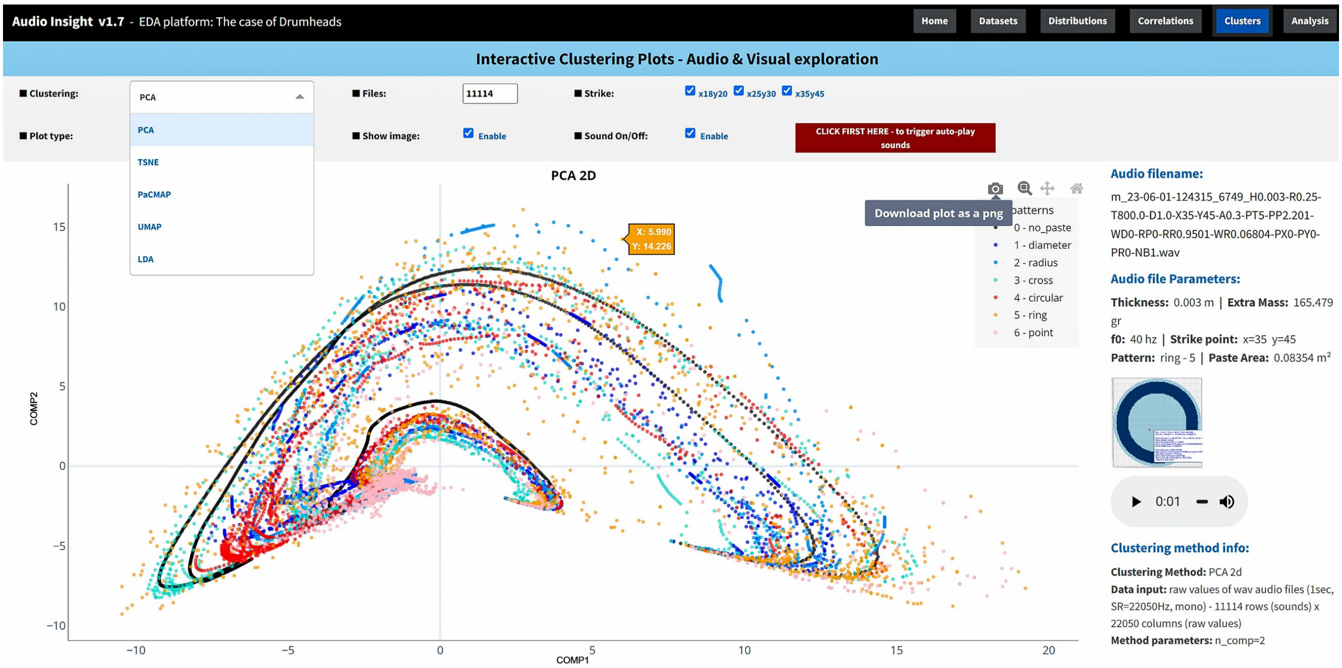

Fig. 6. Clusters page. Points represent sounds. Hovering on different points triggers playback and display of descriptive parameters.

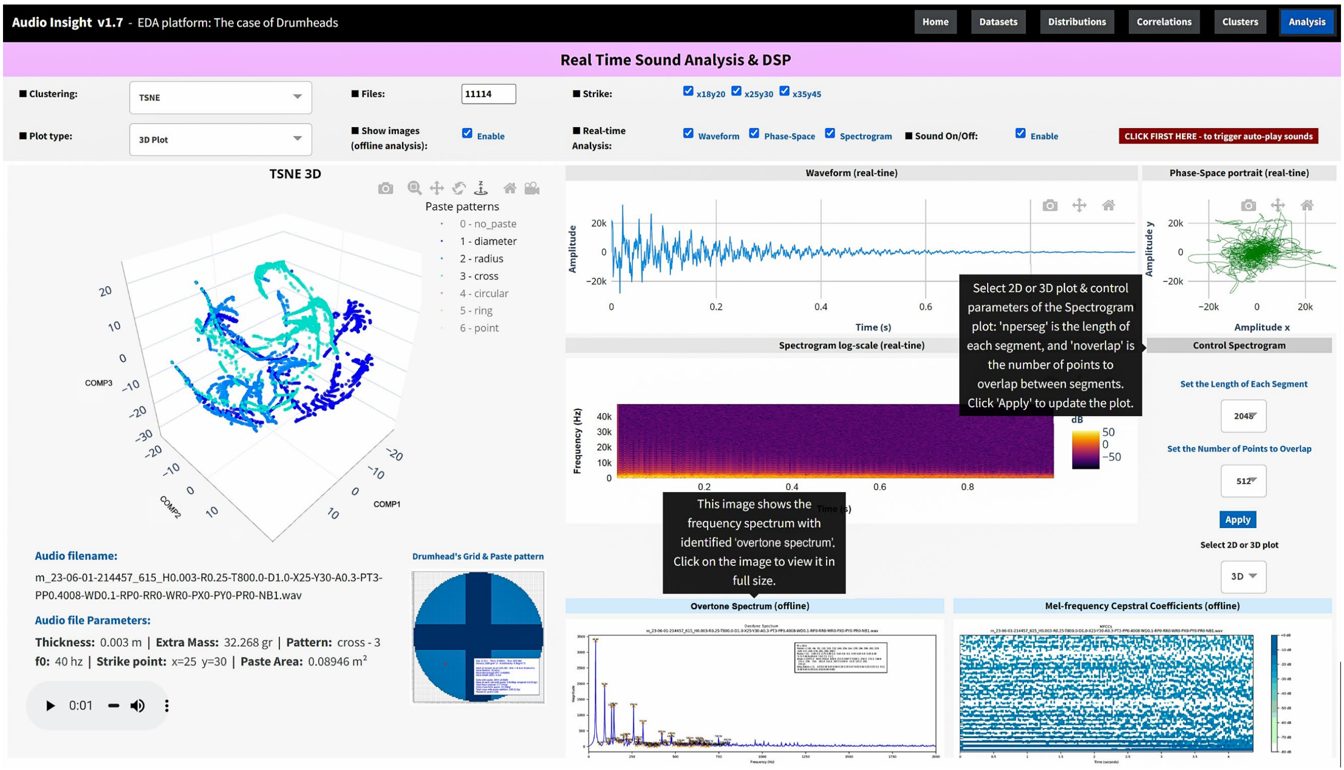

Fig. 7. Analysis Page. t-SNE clustering plot area showing the diameter, radius and cross paste patterns points (left section), along with an offline and real-time audio analysis panel (right section).

E. Analysis Page

The Analysis page (Fig. 7) is designed as a comprehensive tool for real-time sound analysis and digital signal processing. It combines a visualization of data clusters with a set of spectral analysis graphs. The page is split into two sections: a clustering plot area on the left and an analysis panel on the right.

Similarly to the Clusters page, the interactive clustering plot allows for audio playback, while simultaneously performing audio analysis of the corresponding sound file. The analysis panel displays the sound waveform, its Short Term Fourier Transform (STFT) spectrogram and a phase-space portrait visualization that update in real-time as users explore different data points on the cluster area. Additionally, the analysis panel features a sound overtone visualization i.e. an FFT spectrum overlaid by peak information, attained via a peak-picking algorithm, as well as an MFCC spectrogram. These visualizations are generated offline, i.e. prior to loading the dataset to the application.

V. IMPLEMENTATION OF AUDIOINSIGHT

AudioInsight is currently available online without any restrictions related to user authentication. It uses Python as the core programming language and the Dash framework for web development, granting the creation of a responsive web application designed to efficiently handle large audio datasets, while providing a smooth, interactive user experience.

The backend of the application is responsible for data management, audio processing, and data clustering. The frontend is organized into multiple responsive pages, each dedicated to a specific aspect of EDA. Throughout the application, integrated interactive elements such as hover-over audio playback and data filtering significantly enhances the experience of the users in exploring the dataset and identifying potential problems.

A. Dash Framework and Web Development

The Dash framework, developed by Plotly, is an open-source framework specifically designed for building analytical web applications in Python. The architecture of Dash is built on a client-server model, where the server-side component communicates with the client via JSON packets allowing dynamic updates and interactivity. The server uses Flask[16], a lightweight Web Server Gateway Interface (WSGI) framework, to handle incoming HTTP requests and responses, manage URL routing, and provide the necessary functionality to serve web pages.

On the client side, Dash uses React.js[17], a popular JavaScript library for building user interfaces, particularly suited to dynamic web applications. For rendering graphs and visualizations, Dash employs Plotly.js[18], which is a high-level, interactive visualization library. This combination lets Dash create fast, interactive visualizations that can be easily customized and extended. Besides a variety of Plotly visualizations, Dash provides support for Pandas dataframes and the ability to create custom interactive components. It also includes a big collection of pre-made components

[16] https://palletsprojects.com/p/flask/
[17] https://react.dev/
[18] https://github.com/plotly/plotly.js

(Dash Core and HTML Components) for building user interfaces and rendering HTML in Python.

AudioInsight leverages these Dash features to create a responsive and interactive user interface. Dash Bootstrap Components and CSS are used to ensure a responsive layout, capable of adjusting to different screen sizes. Each page is implemented as a separate Dash callback, enabling easier maintenance through a modular development approach. These callbacks handle user interactions and data flow between the frontend and backend of the application.

This modular design, which uses separate Python files for each page, and additional modules for audio feature extraction, plot generation and data processing, ensures easy maintenance and scalability. Furthermore, it allows for extending the capabilities of the application to handle sound datasets that are both generic and tailored to specific problems of musical acoustics.

B. Data Management and Audio Processing

AudioInsight manages its data by efficiently storing and retrieving audio files and related information. This is achieved using the Pandas library for data manipulation and analysis, with a combination of file system storage for the audio files and a curated (labeled) dataset for quick access to metadata and pre-computed features. To increase the response time of the page, the lru_cache decorator from functools[19] is used to optimize performance by caching the results of expensive function calls, particularly when loading and processing data.

In terms of audio processing, AudioInsight uses Scipy, NumPy and Librosa libraries. These modules handle tasks like audio loading, processing and analyzing data, audio feature extraction, and computations for plot generation. To create data visualizations, like scatter plots and histograms in 2D and 3D spaces, the Plotly library is extensively used. It integrates seamlessly with Dash, enabling the creation of dynamic and interactive charts. Clustering algorithms from Scikit-Learn, including PCA, t-SNE, and LDA, as well as modules for PaCMAP and UMAP algorithms, are implemented to provide various trajectories on the dataset structure, and further enhance the analytical capabilities of the application.

C. Status and future perspectives

Currently, the application provides EDA capabilities for our drumhead dataset [30]. However, from the early stages of development and in alignment with our ongoing research in data-driven musical acoustics, it was designed to be extendable to support additional sound datasets without the need to re-engineer the exploratory functionalities. The goal is to make it easy to load new sound datasets with minimal effort and without requiring coding expertise. We are currently in the process of additionally accommodating a dataset of over 200,000 cymbal sounds from well-known cymbal manufacturers. During this process, we are developing a protocol allowing to define data hierarchies and parameter mappings through JSON files. This future integration could lead to the creation of a universal classification system based on measurable sound characteristics, untainted by personal biases and misinterpretations.

A further extension of the AudioInsight application focuses on providing functionalities for extending audio datasets through crowdsourcing and making them available to musicians and musical instrument manufacturers through dedicated user interfaces. Moreover, the AudioInsight may serve as an example for sound texture exploration and development of Smart Musical Instruments (SMIs) [38]. By leveraging the embedded intelligence and sensor data of SMIs, AudioInsight can support real-time crowdsourcing, facilitating the expansion of audio datasets in ways not possible with acoustic instruments. This collaborative approach, combined with principles for guiding the design of SMIs, enables the creation of an interoperable ecosystem where musicians, developers, and manufacturers can work together, enhancing both the scope of the datasets and the overall musical experience.

VI. Discussion and Conclusions

AudioInsight presents an example of a web application aiding research in data-driven musical acoustics. Its development was motivated by our research activities on an inverse acoustic problem, that of computationally inferring how to damp or tune a membrane to produce a desired sound texture. Such computational problems require large amounts of data that are difficult to handle and challenging to assess in terms of accuracy, adequacy and balance to effectively address the required task.

The article does not present any user evaluation other than our own experience to address our research challenges for data exploration. Through the automatic rendering of histograms for parameter distributions, scatter plots for parameter correlations and interactive clustering visualizations, the current version of the application allowed gaining a deeper understanding of our data by: a) revealing areas in which data was sparse, congested, or inaccurate, and b) uncovering complex, non-linear relationships between physical properties and perceptual features that might otherwise remain hidden within the dataset. By (a) revealing problems, researchers are guided to augment or reduce the dataset, for example, by providing more data, by leveraging data augmentation techniques or simply by rejecting certain instances of parameter values as faulty outliers. Through (b) uncovering relationships, researchers are assisted in making research hypotheses that may dictate the need for alternative data representations, for example audio features, to guide analytical processes for data modelling. Nevertheless, besides research-oriented tasks, an outstanding feature of the application is the multisensory navigation of data clusters, which can allow musicians (in this case percussionists), and instrument manufacturers to find out interesting sound textures and gain information on how to reproduce them.

An additional key feature currently being developed is the ability to upload new sounds to the dataset. This functionality will allow researchers to effectively expand the dataset and

[19] https://docs.python.org/3/library/functools.html

musicians to locate their sound signatures within large datasets of registered sounds having pre-annotated physical properties. Such collaborative features could transform AudioInsight into a central hub for researchers, musicians and instrument manufacturers, helping them to share knowledge and work together on investigating perceptually informed physical and structural modifications of musical instrument design.

REFERENCES

[1] L. Turchet et al., "The Internet of Sounds: Convergent Trends, Insights, and Future Directions," in IEEE Internet of Things Journal, vol. 10, no. 13, pp. 11264-11292, 1 July1, 2023, doi: 10.1109/JIOT.2023.3253602

[2] R. Bader, "Finite-Difference model of Mode shape changes of the Myanmar pat wain drum circle using tuning paste," In Proc. Meetings on Acoustics 2016; AIP Publishing: New York, NY, USA, 2016; Volume 29, p. 035004

[3] J.W. Tukey, Exploratory Data Analysis. Reading, MA: Addison-Wesley, 1977.

[4] J.T. Behrens, "Principles and Procedures of Exploratory Data Analysis," Psychological Methods, vol. 2, no. 2, pp. 131-160, 1997, doi: 10.1037/1082-989X.2.2.131.

[5] C. Ahlberg and B. Shneiderman, "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1994.

[6] H. Hochheiser and B. Shneiderman, "Interactive Exploration of Time Series Data," Proceedings of the Fourth International Conference on Information Visualisation, 2003

[7] H. L. M. Lam, "Visual exploratory analysis of large data sets : evaluation and application," T, University of British Columbia, 2008.

[8] C. A. Steed, D. M. Ricciuto, G. Shipman, B. Smith, P. E. Thornton, D. Wang, X. Shi, and D. N. Williams, "Big data visual analytics for exploratory earth system simulation analysis," Comput. Geosci., vol. 61, pp. 71-82, 2013. doi: 10.1016/j.cageo.2013.07.025.

[9] P. Smith, "Exploratory Data Analysis in Climate Science," Int. J. Climatol., vol. 34, no. 2, pp. 345-360, 2016.

[10] S. Yousefzadeh, "An integrated approach for understanding global earthquake patterns and enhancing seismic risk assessment," Int. J. Inf. Technol., vol. 56, pp. 120-134, 2019. https://link.springer.com/article/10.1007/s41870-018-0220-3

[11] S. Filipović, M. Radovanović, and V. Golušin, "Macroeconomic and political aspects of energy security – Exploratory data analysis," Renew. Sustain. Energy Rev., vol. 96, pp. 421-431, 2018. doi: 10.1016/j.rser.2018.08.058.

[12] J. Rahnenführer, R. De Bin, A. Benner, et al., "Statistical analysis of high-dimensional biomedical data: a gentle introduction to analytical goals, common approaches and challenges," BMC Med, vol. 21, article 182, 2023. DOI: 10.1186/s12916-023-02858-y.

[13] A. E. Carpenter, T. R. Jones, M. R. Lamprecht, C. Clarke, I. H. Kang, O. Friman, D. A. Guertin, J. H. Chang, R. A. Lindquist, J. Moffat, P. Golland, and D. M. Sabatini, "CellProfiler: image analysis software for identifying and quantifying cell phenotypes," Genome Biology, vol. 7, no. 10, R100, 2006. doi: 10.1186/gb-2006-7-10-r100.

[14] A. Rule, A. Tabard, and J. D. Hollan, "Exploration and Explanation in Computational Notebooks," in Proc. 2018 CHI Conf. Human Factors Comput. Syst., 2018, doi: 10.1145/3173574.3173606.\

[15] X. Li, Y. Zhang, J. Leung, C. Sun, and J. Zhao, "EDAssistant: Supporting Exploratory Data Analysis in Computational Notebooks with In-Situ Code Search and Recommendation," arXiv preprint, 2021, doi: 10.1145/3411764.3445048.

[16] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster Computing with Working Sets," in Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing (HotCloud'10), USENIX Association, 2010.

[17] L. Battle and J. Heer, "Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau," Computer Graphics Forum, vol. 38, no. 3, pp. 145-159, 2019, doi: 10.1111/cgf.13678.

[18] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer, "Vega-Lite: A Grammar of Interactive Graphics," IEEE Transactions on Visualization and Computer Graphics, vol. 23, no. 1, pp. 341–350, 2017, doi: 10.1109/TVCG.2016.2599030.

[19] K. Wongsuphasawat et al., "Voyager: Exploratory Analysis via Faceted Browsing of Visualization Recommendations," IEEE Transactions on Visualization and Computer Graphics, vol. 22, no. 1, pp. 649–658, Jan. 2016, doi: 10.1109/TVCG.2015.2467191.

[20] C. Cannam, C. Landone, and M. Sandler, "Sonic Visualiser: An Open Source Application for Viewing, Analysing, and Annotating Music Audio Files," in Proc. 18th ACM Int. Conf. Multimedia (MM '10), New York, NY, USA, 2010, pp. 1467–1468, doi: 10.1145/1873951.1874248.

[21] B. McFee, C. Raffel, D. Liang, D. P. W. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and Music Signal Analysis in Python," in Proceedings of the 14th Python in Science Conference, 2015, pp. 18-25, doi: 10.25080/Majora-7b98e3ed-003

[22] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "Essentia: An Audio Analysis Library for Music Information Retrieval," in Proc. of the 14th International Society for Music Information Retrieval Conference (ISMIR), Curitiba, Brazil, Nov. 4-8, 2013.

[23] E. Bakhturina, V. Lavrukhin, and B. Ginsburg, "A Toolbox for Construction and Analysis of Speech Datasets," arXiv: 2104.04896. [Online] Available: https://arxiv.org/abs/2104.04896

[24] V. Dibia and C. Demiralp, "Data2Vis: Automatic Generation of Data Visualizations Using Sequence-to-Sequence Recurrent Neural Networks," IEEE Comput. Graph. Appl., vol. 39, no. 5, pp. 33–46, Sep. 2019, doi: 10.1109/MCG.2019.2924636.

[25] Y. Luo, X. Qin, N. Tang, and G. Li, "DeepEye: Towards Automatic Data Visualization," in Proc. Int. Conf. Data Eng., 2018, doi: 10.1109/ICDE.2018.00019

[26] K. Hu, M. A. Bakker, S. Li, T. Kraska, and C. Hidalgo, "VizML: A Machine Learning Approach to Visualization Recommendation," in Proc. 2019 CHI Conf. Human Factors Comput. Syst., 2019, doi: 10.1145/3290605.3300358

[27] O. Bar El, T. Milo, and A. Somech, "Automatically Generating Data Exploration Sessions Using Deep Reinforcement Learning," in Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (SIGMOD '20), New York, NY, USA, Association for Computing Machinery, 2020, pp. 1527-1537. doi: https://doi.org/10.1145/3318464.3389779.

[28] D. Heise and H. L. Bear, "Visually Exploring Multi-Purpose Audio Data," arXiv preprint arXiv:2110.04584, 2021. [Online]. Available: https://arxiv.org/abs/2110.04584.M. Olivieri et al., "Audio Information Retrieval and Musical Acoustics," IEEE Instrum. Meas. Mag., vol. 24, no. 7, pp. 10-20, 2021, doi: 10.1109/MIM.2021.9549233.

[29] P. Fallgren, Z. Malisz, and J. Edlund, "A tool for exploring large amounts of found audio data," in CEUR Workshop Proceedings, 3rd Conference on Digital Humanities in the Nordic Countries (DHN 2018), 7-9 March 2018, pp. 499-503.

[30] C. Alexandraki, M. Starakis, P. Zervas, and R. Bader, "Inferring Drumhead Damping and Tuning from Sound Using Finite Difference Time Domain (FDTD) Models," Acoustics, vol. 5, no. 3, pp. 798-816, 2023. DOI: 10.3390/acoustics5030047.

[31] C. Alexandraki, M. Starakis, R. Bader, and P. Zervas, "Machine learning of Finite-Difference Time Domain (FDTD) physical modelling sound simulations of drumhead paste pattern distributions," in Proc. European Acoustics Association Forum Acusticum 2023, Torino, Italy, Sept. 2023, pp. 2217-2224. DOI: 10.61782/fa.2023.1053

[32] T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 374, no. 2065, p. 20150202, Apr. 2016, doi: 10.1098/rsta.2015.0202.

[33] G. Hinton and S. Roweis, "Stochastic neighbor embedding," in Proceedings of the 15th International Conference on Neural Information Processing Systems (NIPS'02), Cambridge, MA, USA, 2002, pp. 857-864. MIT Press.

[34] L. van der Maaten and G. E. Hinton, "Visualizing Data using t-SNE," Journal of Machine Learning Research, vol. 9, pp. 2579-2605, 2008.

[35] Y. Wang, H. Huang, C. Rudin, and Y. Shaposhnik, "Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization," Dec. 2020

[36] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," arXiv, 2020. [Online]. Available: https://arxiv.org/abs/1802.03426.

[37] R. A. Fisher, "The use of multiple measurements in taxonomic problems," Annals of Eugenics, vol. 7, pp. 179-188, 1936. doi: http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x.

[38] L. Turchet, "Smart Musical Instruments: Vision, Design Principles, and Future Directions," IEEE Access, vol. 7, pp. 8944-8963, 2019, doi: 10.1109/ACCESS.2018.2876891.