



Vol. 3 N.º 9 / Guayaquil
mayo 2024
ISSN 2953-657x

La sonificación de datos como herramienta tecnológica para la creación artística

Diego Alejandro Molina Quintero

Facultad de Artes y Humanidades
Instituto Tecnológico Metropolitano
Medellín, Colombia
diegomolina4406@correo.itm.edu.co

Isabella Álvarez Franco

Facultad de Artes y Humanidades
Instituto Tecnológico Metropolitano
Medellín, Colombia
isabellaalvarez251817@correo.itm.edu.co

Lina Alejandra González Ramírez

Facultad de Artes y Humanidades
Instituto Tecnológico Metropolitano
Medellín, Colombia
linagonzalez305094@correo.itm.edu.co

Jamir Mauricio Moreno Espinal

Facultad de Artes y Humanidades
Instituto Tecnológico Metropolitano
Medellín, Colombia
mauriciomoreno@itm.edu.co

Resumen

La sonificación de datos (en adelante SD) es una técnica cada vez más utilizada en diversos campos para convertir bases de datos en sonidos y permitir su interpretación auditiva. Para tal propósito, en este estudio se utilizaron los softwares TwoTone, SuperCollider y Sonic Visualiser. De esa forma, el artista puede elegir la herramienta de creación más adecuada a sus intereses y capacidades para lograr las experiencias sonoras deseadas.

Esta investigación aplicó los métodos analítico y comparativo, ya que examina en detalle la forma de uso de las herramientas seleccionadas, su proceso de instalación y ejecución, y explora la personalización de los sonidos y su integración con dispositivos externos como, por ejemplo, placas de desarrollo similares a Arduino o Raspberry Pi. Adicionalmente, realiza un análisis comparativo de las estadísticas de las contribuciones realizadas por la comunidad en GitHub para cada una de estas herramientas.

Los resultados obtenidos pueden tener aplicaciones en campos como la música, la ciencia de datos y la investigación en sonido. Además, pueden ser de gran utilidad para los usuarios interesados en realizar sonificaciones de manera creativa y personalizada.

Es importante destacar que esta investigación se centró en explorar la generación de sonidos simples y básicos en cada plataforma evaluada. Sin embargo, se debe recordar que es posible que la elección de la herramienta de software para la sonificación de datos dependa en gran medida del nivel de conocimiento del usuario en áreas como la programación, la música y el manejo de datos. Estos factores influyen en la capacidad del usuario para aprovechar al máximo las herramientas disponibles y personalizar los resultados según sus necesidades específicas. Para aquellos con un nivel bajo de conocimiento, TwoTone y Sonic Visualiser pueden ser una buena opción, debido a su intuitiva interfaz de usuario. Por su parte, SuperCollider es una herramienta poderosa para usuarios con conocimientos medios y altos en programación, pero su curva de aprendizaje es empinada. Respecto a la personalización, SuperCollider y TwoTone ofrecen una amplia variedad de opciones, mientras que Sonic Visualiser se enfoca principalmente en el análisis y la visualización de datos de audio. Cada herramienta tiene, pues, sus limitaciones y fortalezas en relación con la generación y la personalización de sonidos.

Palabras clave: sonificación, programación creativa, TwoTone, SuperCollider, Sonic Visualiser

Abstract

Data sonification, hereafter referred to as SD, is an increasingly utilized technique in various fields to convert databases into sounds and enable auditory interpretation. In this study, the software tools TwoTone, SuperCollider, and Sonic Visualiser were employed for this purpose. In this way, artists can choose the creation tool most appropriate to their interests and abilities to achieve the desired sound experiences.

This research employed analytical and comparative methods to examine in detail the usage of each selected tool, including their installation and execution processes, as well as exploring sound customization and integration with external devices such as Arduino or Raspberry Pi development boards. Additionally, a comparative analysis of contribution statistics from the GitHub community was conducted for each of these tools.

The results have potential applications in fields like music, data science, and sound research, offering significant value to users interested in creatively and personally generating sonifications.

The choice of software tool for sonifications depends on the user's proficiency in programming, music, and data. For those with limited knowledge, TwoTone and Sonic Visualiser are recommended due to their intuitive user interface. On the other hand, SuperCollider is a powerful tool suitable for users with intermediate to advanced programming skills, despite its steep learning curve. In terms of customization, SuperCollider and TwoTone provide a wide range of options, while Sonic Visualiser primarily focuses on audio data analysis and visualization. Therefore, each tool has its own limitations and strengths concerning sound generation and customization.

Keywords: sonification, creative programming, TwoTone, SuperCollider, Sonic Visualiser

1. Introducción

La sonificación es una técnica que permite transformar información en sonidos.¹ A pesar de que no es tan conocida ni se practica tanto como la visualización gráfica de datos, el sonido, en todo caso, es un medio naturalmente afectivo, estético y cultural.² Aunque parezca una idea moderna, la humanidad lo ha utilizado desde siglos atrás para interpretar información; un ejemplo de ello son las campanas de las iglesias para marcar el tiempo y convocar a la congregación, sin dejar a un lado su papel para señalar el horario de las actividades seculares.³

El sonido también se ha usado para representar datos en campos como la detección de la radiación, la distancia de los objetos en el mar, la altura de los aviones o las afectaciones en la salud. Es así como el contador Geiger, un dispositivo que utiliza un tubo Geiger-Müller para medir la cantidad de partículas ionizantes en el aire⁴ produce, según su cantidad, un sonido que advierte que su concentración es alta y puede ser peligrosa para el ser humano.

Otro ejemplo destacado del uso del sonido para representar datos es el optófono, creado en 1913 para facilitarles la lectura a las personas invidentes. Actualmente, la tecnología ha permitido el desarrollo de múltiples dispositivos de este tipo para individuos con discapacidad visual, a través

1 David Rodríguez Mateos y Alicia Tapia López, «Sonificación y periodismo: la representación de datos mediante sonidos», *Revista de Comunicación*, vol. 22, n.º 1 (2023): 413–433, <http://dx.doi.org/10.26441/rc22.1-2023-3022>

2 Paul Vickers y Bennet Hogg, «Sonification abstraite / Sonification concrète: An aesthetic perspective space for classifying auditory displays in the Ars Musica domain», en *Proceedings of the 12th International Conference on Auditory Display*, (210–216) (Londres, 2006). <https://typeset.io/pdf/sonification-abstraite-sonification-concrete-an-aesthetic-1urt1pe0ql.pdf>

3 Anne Staples, «El abuso de las campanas en el siglo pasado», *Historia Mexicana*, vol. 27, n.º 2 (1977): 177–194, <https://www.jstor.org/stable/25135599>

4 El tubo Geiger-Müller es un tubo con gas que contiene un electrodo positivo y uno negativo. Leonardo Ramírez Arias y Julián Uribe Cortés, «El contador Geiger como herramienta en la enseñanza del concepto de radiación», *Góndola, Enseñanza y Aprendizaje de las Ciencias*, vol. 2, n.º 1 (2007): 33–36, <https://revistas.udistrital.edu.co/index.php/GDLA/article/download/5304/6935/0>

de tecnologías como la OCR (*optical character recognition*) y el TTS (*text-to-speech*).⁵

Entre 1929 y 1930 surgieron herramientas como el radioaltímetro y el sonar. El primero utiliza el radar para emitir pulsos de ondas hacia el suelo,⁶ y el segundo se usa como un medio de localización acústica de forma similar al radar. Sin embargo, este último, en lugar de emitir ondas electromagnéticas, emplea impulsos sonoros.⁷ Cabe anotar que, anterior a la aparición del GPS (Global Positioning System), la localización acústica se había usado en el aire para la investigación atmosférica⁸ a través de la aplicación del Sodar (Sonic Detection and Ranging, exploración vertical aérea con el sonar). Con esta tecnología, la interpretación de los datos verticales podía guiar a las aeronaves emitiendo un sonido de alerta si frente a ellas había montañas u obstáculos amenazadores de su seguridad.

En 1940 se desarrolló el ecógrafo, que utiliza el ultrasonido para visualizar estructuras tan pequeñas que escapan a la detección del ojo humano.⁹ Este dispositivo ha sido utilizado en la medicina durante muchos años, facilitando la auscultación de los ruidos cardíacos o respiratorios, así como para objetivar ruidos intestinales o soplos por flujos anómalos sanguíneos en venas y arterias.¹⁰

5 I. A. J. Gárate Piñones, «Doroti» (tesis doctoral, Universidad Andrés Bello, 2020).

6 Bryan Marcelo Peralta Rocha, «Inspección y pintura del avión Escuela Hawker Siddeley 125-400, mediante información y datos técnicos para la unidad de gestión de tecnologías UGT-ESPE» (tesis de pregrado, Universidad de las Fuerzas Armadas, Sangolquí, Ecuador, 2018), <https://repositorio.espe.edu.ec/bitstream/21000/27745/1/M-ESPEL-CMA-0478.pdf>

7 «Sonar», Wikipedia, 21 de marzo de 2019, <https://es.wikipedia.org/wiki/Sonar>

8 Andrea Vicenta Remiro Gargallo, «Técnicas numéricas aplicadas a la propagación de sonido en el medio marino» (tesis doctoral, Universitat Politècnica de València, España, 2019), <http://hdl.handle.net/10251/128636>

9 Andrea del Pilar Sánchez Corzo, «Sonorización de señales de EEG basada en estructuras musicales» (tesis de pregrado, Universidad de los Andes, Bogotá, 2012), <https://repositorio.uniandes.edu.co/bitstream/handle/1992/25182/u627972.pdf?sequence=1&isAllowed=y>

10 Fernando Saraví, «Laennec, el método anamoclinico y la invención del estetoscopio. Parte 1: De Bretaña a París, hasta la gran invención», *Revista Médica Universitaria*, vol. 8, n.º 2 (2012), https://bdigital.uncu.edu.ar/objetos_digitales/4769/rmuhistoriasdelamedicina.pdf

Por otro lado, el estetoscopio es un dispositivo que ha ayudado a la medicina por largo tiempo, facilitando la escucha interna del cuerpo.¹¹ Este instrumento se utiliza de la misma manera que el ecógrafo.¹²

Finalmente, los detectores de metales, que emiten sonidos cuando localizan metales a diferentes profundidades, tienen una gran variedad de aplicaciones en la industria actual tanto en materia de seguridad como en el ocio.¹³

En resumen, la sonificación, una técnica utilizada por la humanidad desde hace siglos, permite transformar información en sonidos. A lo largo de la historia se han desarrollado dispositivos que utilizan el sonido para representar datos en campos como la medicina, la detección de la radiación o la exploración aérea o marina. En la actualidad, gracias a la tecnología, se han desarrollado múltiples dispositivos que utilizan el sonido para facilitar la vida de las personas.

Este documento profundiza en aspectos técnicos de las herramientas de software evaluadas, como la capacidad de SuperCollider para la síntesis de sonidos complejos y su curva de aprendizaje más empinada, o las interfaces intuitivas y fáciles de usar de TwoTone y Sonic Visualiser, que las hacen más accesibles para aquellos usuarios con menores conocimientos en programación y música. Además, se pueden explorar las limitaciones y fortalezas de cada herramienta en términos de su procesamiento de las señales y el análisis y la visualización de los datos de audio, lo que puede afectar la calidad y la complejidad del sonido generado.

2. Perspectiva teórica

La importancia del sonido para el ser humano es indiscutible. Más allá de la vista, el

11 «Estetoscopio», Quimica.es, <https://www.quimica.es/enciclopedia/Estetoscopio.html>

12 Saraví, «Laennec, el método...».

13 Carlos Martínez Cantón, «Diseño y construcción de un detector de metales basado en un microcontrolador» (tesis de pregrado, Universitat Politècnica de Catalunya, España, 2016), <https://upcommons.upc.edu/handle/2117/98926?locale=attribute=es>

olfato, el tacto y el gusto, la audición es, sin duda, una de las funciones vitales para sobrevivir e interactuar con el mundo exterior. El oído tiene la capacidad de transformar las ondas físicas en impulsos nerviosos que llegan al cerebro, permitiéndole a las personas interactuar con el mundo desde temprana edad y desarrollar diferentes habilidades. El oído es esencial, ya que la humanidad se desenvuelve en un paisaje sonoro que le permite desarrollar diferentes tipos de habilidades.¹⁴

Los sonidos tienen un papel clave en la experiencia sensorial y emocional del mundo que rodea al ser humano, permitiéndole distinguir diferentes timbres y procesarlos de forma fluida. De hecho, los sonidos tienen un rango de intensidades mucho mayor que las detectadas por la vista humana para percibir diferentes niveles de luz¹⁵ y, por tanto, lo influyen constantemente y generan diferentes reacciones en su cerebro.

A través del oído, el humano es capaz de procesar diferentes tipos de sonidos, interpretarlos y reaccionar ante ellos, lo que le brinda una amplia variedad de habilidades y le permite tener una experiencia sensorial y emocional más completa. Además, la SD le permite traducir la información en sonidos para una mejor comprensión auditiva que pueda aplicarse en áreas como la seguridad, la monitorización y el arte, esta última, en particular, por su valor estético y artístico. La importancia del sonido en la vida de la humanidad es indudable, por lo que es fundamental seguir valorando y explorando esta capacidad sensorial para mejorar la comprensión y la experiencia del mundo que la rodea.

A continuación, se presenta un listado de los programas que permiten la SD.

14 Pierre Schaeffer, Christine North y John Dack, *Treatise on Musical Objects: An Essay Across Disciplines* (California: University of California Press, 2017), <https://www.jstor.org/stable/10.1525/j.ctt1qv5pb>

15 Alina Castillo Vázquez, Diedmar Céspedes Cabrera, Mercedes Jiménez Betancourt y Gustavo Rodríguez Rosabal, «Alternativa didáctica para el estudio de los órganos sensitivos en la enseñanza de la Biología 3», *Revista Dilemas Contemporáneos: Educación, Política y Valores*, vol. 8, n.º 1 (2020), <https://dilemascontemporaneoseducacionpoliticaayvalores.com/index.php/dilemas/article/view/2386>

- TwoTone: *software* gratuito y de código abierto que se ejecuta totalmente en la red y no requiere ser descargado. TwoTone permite convertir datos en sonidos y música a través del proceso de sonificación, y es compatible con computadoras de escritorio, tabletas y teléfonos móviles. Además, es una herramienta útil tanto para la comprensión de datos como para la creación de música, sin necesidad de conocimientos musicales o tecnológicos previos.¹⁶
- Pure Data (Pd): lenguaje de programación gráfico desarrollado por Miller Puckette para la creación de música interactiva y obras multimedia. Pd es un entorno de programación visual de código abierto que se puede ejecutar en una variedad de dispositivos, desde computadoras personales hasta unidades integradas y teléfonos inteligentes. Este *software* permite a músicos, artistas visuales, investigadores y desarrolladores crear *software* gráficamente sin tener que escribir líneas de código. Pd se puede utilizar para procesar y generar sonido, video, gráficos en 2D y 3D, sensores de interfaz, dispositivos de entrada y MIDI (Musical Instrument Digital Interface).¹⁷
- SuperCollider: *software* libre desarrollado por James McCartney en 1996, que presenta tres componentes principales: un servidor de audio en tiempo real, un lenguaje de programación interpretado y un editor de scLang con un sistema de ayuda integrado. SuperCollider es adecuado para aprender métodos básicos de procesamiento multimedia y programación visual, así como para realizar sistemas complejos para proyectos a gran escala.¹⁸
- Dear Diary ai: sitio web que permite a los usuarios traducir lo que se escribe en sonidos. Es una herramienta útil para las personas que deseen experimentar con sonidos a partir de textos.¹⁹
- Sonic Visualiser: *software* libre y de código abierto que permite a los usuarios la visualización y el análisis de archivos de sonido. Adicionalmente, ofrece funciones de sonificación que hacen posible convertir los datos de un archivo de sonido en un sonido audible.²⁰
- Sonification Sandbox: plataforma en línea que permite a los usuarios cargar y sonificar datos de diferentes tipos. Adicionalmente, cuenta con herramientas de análisis de sonido y facilita la descarga de archivos de audio.²¹
- AudioSpray: *software* que permite a los usuarios la creación de sonidos y música a partir de imágenes. Es una herramienta útil para la creación de paisajes sonoros y efectos de sonido.²²
- Sonic Pi: *software* libre y gratuito que permite a los usuarios crear música electrónica mediante programación. El programa cuenta con un entorno de desarrollo interactivo en el que pueden escribir código en lenguaje Ruby para crear y manipular sonidos en tiempo real. Además,

16 Gonzalo de la Vega, Leonardo Martín Exequiel Domínguez, Johanna Casado, Beatriz García, «SonoUno Web: An innovative user centred web interface», *Arxiv* (2022), <https://arxiv.org/pdf/2302.00081.pdf>

17 Francisco Javier Ruiz-del-Olmo y José Vertedor, «Procesos, herramientas y prácticas de la sonificación», Málaga, España, Universidad de Málaga, 2016, <http://hdl.handle.net/10630/12429>

18 Scott Wilson, David Cottle y Nick Collins, *The SuperCollider Book* (Cambridge, MA: The MIT Press, 2011).

19 Hengbo Wang, «Dear diary: an investigation of emotionally engaging computer-game narratives» (tesis de maestría, Auckland University of Technology, Australia, 2015), <http://hdl.handle.net/10292/9137>

20 Chris Canam, Christian Landone, Mark Sandler y Juan Pablo Bello, «The sonic visualiser: a visualisation platform for semantic descriptors from musical signals», *ISMIR 2006 7th International Conference on Music Information Retrieval*, 324–327 (Victoria, BC, Canadá: 2006), <https://nyuscholars.nyu.edu/en/publications/the-sonic-visualiser-a-visualisation-platform-for-semantic-descri>

21 Bruce N. Walker y Joshua T. Cothran, «Sonification sandbox: a graphical toolkit for auditory graphs», *Proceedings of the 2003 International Conference on Auditory Display* (Atlanta, GA: Georgia Institute of Technology, 2003).

22 Richard Garrett, «AudioSprayGun0.8-The generation of large sound-groups and their use in three-dimensional spatialisation», *ICMC* (2015), <https://quod.lib.umich.edu/cgi/p/pod/dod-idx/audio-spray-gun-08-the-generationof-large-sound-groupspdf?c=icmc;idno=-bbp2372.2015.071;format=pdf>

incluye una amplia variedad de muestras y sonidos predefinidos para facilitar el proceso de creación musical.²³

- Max/MSP: entorno de programación visual para la creación de música, sonido y multimedia. Max/MSP es ampliamente utilizado por músicos, artistas visuales, diseñadores de sonido y desarrolladores de software para crear instalaciones interactivas, performances y composiciones multimedia. El programa permite a los usuarios crear sus propios objetos y algoritmos personalizados para procesar y generar sonidos, video, gráficos y otros tipos de datos.²⁴
- Csound: lenguaje de programación para la síntesis de sonido y el procesamiento de señales de audio. Csound ha sido utilizado por músicos, compositores y artistas sonoros para crear una amplia variedad de obras musicales y experimentales. El programa permite a los usuarios definir y manipular múltiples canales de audio, crear efectos de procesamiento de señales y controlar dispositivos externos a través de MIDI.²⁵

23 Aaron, S., «Sonic Pi-The live coding music synth for everyone», Sonic Pi, 2018, <https://sonic-pi.net/>

24 Daniel Edward Barsetti-Nerland, «Creating a responsive visualization that reacts with music in real time: integrating Ableton Live 9 and Cycling'74 max for live into a musical performance» (tesis de pregrado, University of Arizona, Tucson, 2016), <https://repository.arizona.edu/handle/10150/612546>

25 B. Vercoe et al., «The canonical Csound reference manual», Csound, 2007, <https://csound.com/docs/manual/index.html>

- ChucK: lenguaje de programación de audio desarrollado por Ge Wang y Perry Cook para la creación de música en tiempo real y multimedia interactiva. ChucK es un lenguaje de programación textual que permite a los usuarios crear y manipular sonidos utilizando objetos y sintaxis especializados. Además, cuenta con herramientas de análisis y síntesis de sonido que facultan la sonificación de datos y la generación de diferentes tipos de sonidos y efectos.²⁶

La tabla 1 muestra un resumen de los programas SD expuestos.

Tabla I
Programas de sonificación de datos

Software	Sistemas operativos	Github	Creador
TwoTone	Windows, Mac, Linux	https://github.com/sonifydata/twotone	Paul Batchelor
Pure Data (Pd)	Windows, Mac, Linux	https://github.com/pure-data/pure-data	Miller Puckette
Super Collider	Windows, Mac, Linux	https://github.com/supercollider/supercollider	James McCartney
Dear Diary ai	Windows, Mac, Linux	https://github.com/frederick0329/Dear-diary-ai	frederick 0329 (usuario de GitHub)
Sonic Visualiser	Windows, Mac, Linux	https://github.com/svi-project/sonic-visualiser	Chris Cannam
Sonification Sandbox	Windows, Mac	https://github.com/Ching-Chieh/Sonification-Sandbox	Ching-Chieh Cheng
SoniPy	Windows, Mac, Linux	https://github.com/carlthome/python-sonify	Carl Thomé
Audio Spray	Windows	https://github.com/flexmonkey/audiospray	Matt Bellingham
Sonic Pi	Windows, Mac, Linux	https://github.com/samaaron/sonic-pi	Sam Aaron
Max/MSP	Windows, Mac	https://github.com/Cycling74/max-sdk	Miller Puckette
Csound	Windows, Mac, Linux	https://github.com/csound/csound	Barry Vercoe
ChucK	Windows, Mac, Linux	https://github.com/ccrma/chuck	Ge Wang, Perry Cook et al.

Fuente: elaboración de los autores.

26 Ge Wang, Dan Trueman, Scott Smallwood y Perry R. Cook, «The laptop orchestra as classroom», *Computer Music Journal*, vol. 32, n.º 1 (2008): 26–37, <https://www.jstor.org/stable/40072662>

3. Perspectiva metodológica

La metodología empleada en el artículo proporciona una descripción detallada de los procedimientos y pasos seguidos para evaluar y comparar tres herramientas de sonificación de datos: TwoTone, SuperCollider y Sonic Visualiser.

3.1. Fases de la investigación

La investigación se llevó a cabo en varias fases claramente definidas:

- Selección del *software*: se identificaron y seleccionaron las herramientas de sonificación de datos que se estudiarían. Se describieron criterios específicos para la elección de estas herramientas, como la disponibilidad en múltiples sistemas operativos y la accesibilidad del código fuente en GitHub.
- Instalación y configuración: se detallaron los procedimientos de instalación de cada herramienta en un entorno operativo basado en Ubuntu 22.04.2 LTS x86_64. Esto incluyó los comandos y pasos necesarios para la instalación de cada herramienta.
- Caso de uso y personalización: se proporcionó un ejemplo práctico de cómo utilizar cada herramienta para generar sonidos a partir de una base de datos específica. Se explicó cómo personalizar y modificar los sonidos según las necesidades del usuario.

Esta sección presenta la descripción detallada de la implementación de tres herramientas de SD: TwoTone, SuperCollider y Sonic Visualiser.

Dichas herramientas fueron instaladas en un entorno operativo basado en Ubuntu 22.04.2 LTS x86_64, ejecutado

en un equipo modelo Katana GF76 12UC REV:1.0.

El equipo cuenta con las siguientes especificaciones técnicas: procesador de 12.^a generación Intel i7-12700H y una unidad de procesamiento gráfico NVIDIA GeForce RTX 3050 Mobile, y opera con el Kernel 5.19.0-35-generic y la shell Zsh 5.8.1. Adicionalmente, se llevó a cabo una revisión exhaustiva de las principales características y funcionalidades de cada plataforma, usando el método analítico combinado con el método comparativo.

3.1.1. Selección del software

La selección de las herramientas de sonificación de datos desempeña un papel fundamental en cualquier investigación que involucra la creación de sonidos a partir de datos. Para garantizar la idoneidad de las herramientas elegidas y la validez de las evaluaciones realizadas, se establecieron criterios específicos de selección. Estos criterios se basaron en consideraciones clave que afectarían tanto la accesibilidad como la calidad del análisis. Los criterios utilizados para seleccionar las herramientas incluyeron:

- Disponibilidad en múltiples sistemas operativos: se priorizó la selección de herramientas que estuvieran disponibles en una variedad de sistemas operativos ampliamente utilizados, como Windows, Mac y Linux. Esto se hizo con el objetivo de garantizar que las herramientas fueran accesibles para un público diverso y no limitar su utilidad a un sistema operativo específico.
- Accesibilidad del código fuente: se consideró esencial que las herramientas seleccionadas fueran de código abierto y que su código fuente estuviera disponible públicamente a

- través de plataformas como GitHub. Esto facilitaría la transparencia y la capacidad de examinar y evaluar el funcionamiento interno de las herramientas, lo que es esencial para la investigación y la evaluación objetiva.
- Comunidad y documentación activa: se evaluó la presencia de una comunidad activa de usuarios y desarrolladores en torno a cada herramienta. Una comunidad activa es indicativa de un soporte continuo, corrección de errores y desarrollo en curso. Además, se consideró la disponibilidad de documentación detallada y recursos de aprendizaje que facilitaran la comprensión y el uso de las herramientas.
 - Estabilidad y madurez: se prestó atención a la estabilidad y madurez de las herramientas, lo que incluyó la evaluación de la frecuencia de actualizaciones, la ausencia de problemas críticos conocidos y la duración de su existencia en el ámbito de la sonificación de datos.
 - Funcionalidad relevante: se tuvo en cuenta la funcionalidad específica de sonificación de datos de cada herramienta. Esto incluyó la capacidad de generar sonidos a partir de datos, así como la flexibilidad y versatilidad en la manipulación de estos sonidos para adaptarse a diferentes contextos de investigación.
 - Licencia de software: se verificó que las herramientas seleccionadas utilizaran licencias de software compatibles con los objetivos de la investigación y la publicación de resultados. Las licencias de código abierto o de uso gratuito fueron preferidas para promover la accesibilidad y el intercambio de conocimientos.

- Aplicabilidad a la investigación: se consideró si las herramientas eran adecuadas para los objetivos específicos de la investigación, lo que incluyó la capacidad de trabajar con los tipos de datos y bases de datos utilizados en el estudio.

La aplicación de estos criterios aseguró que las herramientas seleccionadas fueran apropiadas y justas para la comparación y evaluación en el contexto de la sonificación de datos. Además, proporcionó una base sólida para la objetividad y la replicabilidad de los resultados de la investigación.

3.1.2. Instalación

TwoTone

Cabe recordar que esta aplicación puede usarse en línea desde cualquier navegador (<https://twotone.io/>), aunque también permite descargar el código y emplearlo *offline* de la siguiente forma:

- Clonar el proyecto en un directorio de la computadora: git clone <https://github.com/datavized/twotone.git>
- Ingresar a la carpeta en la que se acaba de clonar desde la terminal usando el siguiente comando: cd twotone
- Instalar las dependencias necesarias usando el siguiente comando: npm instal.
- Para poder ejecutarlo en modo de desarrollo, se necesita configurar un servidor web local en la computadora. Esto permitirá visualizar la aplicación en el navegador web del usuario sin necesidad de publicarla en un servidor remoto. Además, cuando se realicen cambios en los archivos de código fuente, la aplicación se reconstruirá

- automáticamente para reflejarlos, sin que haya que reiniciar manualmente el servidor; el comando es el siguiente: npm run star
- Para compilar una versión de producción del proyecto, se puede utilizar el siguiente comando: npm run build

SuperCollider

Para la instalación estándar en Ubuntu, se deben ejecutar los siguientes comandos de la terminal, una línea a la vez:

```
sudo add-apt-repository ppa:super-
collider/ppa
sudo apt-get update
sudo apt-get install supercollider
```

Para enviar los datos que se van a procesar y enviarlos a SuperCollider, se utiliza Python. Es importante tener en cuenta las siguientes consideraciones al utilizar Python en Linux:

- Crear un entorno virtual. Un entorno virtual es un espacio donde se pueden instalar paquetes específicos para un proyecto. Esto significa que es posible tener un conjunto de paquetes/librerías aislados de la instalación principal de Python en el sistema para un proyecto en particular.
- Abrir la terminal en Ubuntu Linux e instalar pip (el administrador de paquetes de Python) en el sistema. Si aún no lo está, puede hacerse con el siguiente comando: sudo apt-get install python3-pip
- Instalar virtualenv, una herramienta para crear entornos virtuales de Python, a través del siguiente comando: sudo pip3 install virtualenv
- Crear un nuevo directorio para el entorno virtual, que será su directorio

raíz. Se puede nombrar de cualquier forma, por ejemplo: «mi_entorno_virtual», a través del siguiente comando: mkdir mi_entorno_virtual

- Entrar en el directorio recién creado utilizando el siguiente comando: cd mi_entorno_virtual
- Crear un entorno virtual dentro de este directorio. Se puede nombrar de cualquier forma, por ejemplo: «venv», a través del siguiente comando: virtualenv venv
- Activar el entorno virtual utilizando el siguiente comando: source venv/bin/activate

Ahora, el usuario ya está dentro del entorno virtual y debe instalar el paquete de Python que va a usar utilizando pip a través del siguiente comando: pip install numpy y pip install python-osc

Una vez que SuperCollider esté instalado, puede ejecutarse en la terminal ingresando el siguiente comando: sclang. Esta acción iniciará el intérprete de SuperCollider en la terminal.

Sonic Visualiser

1. Abrir la terminal de Ubuntu y actualizar la lista de paquetes con el siguiente comando: sudo apt-get update
2. Instalar Sonic Visualiser con el siguiente comando: sudo apt-get install sonic-visualiser
3. Esperar a que el proceso de instalación termine. Si se solicita confirmación, presionar «y» y «Enter» para continuar. Una vez que la instalación esté completa, el programa puede abrirse desde el menú de aplicaciones de Ubuntu o desde la terminal, a través del siguiente comando: sonic-visualiser

Tabla II
Aspectos técnicos de instalación, lenguaje, navegabilidad y tipo de licencia de las herramientas seleccionadas

Aspecto	TwoTone	SuperCollider	Sonic Visualiser
Sistema operativo	Windows, Mac, Linux	Windows, Mac, Linux	Windows, Mac, Linux
Lenguajes de programación	JavaScript	C++, Super Collider language	C++, Python
Interfaz gráfica	Sí	No (interfaz de línea de comandos)	Sí
Navegabilidad web	Sí (se puede ejecutar en un navegador web)	No	No
Tipo de licencia	Licencia MIT (software libre y de código abierto)	Licencia GPLv2 (software libre y de código abierto)	Licencia GPL (software libre y de código abierto)

Fuente: elaboración de los autores.

4. Resultados y discusión

TwoTone

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/twotone>

SuperCollider

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/supercollider>

Sonic Visualiser

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/sonicvisualiser>

Esta sección desempeña un papel crucial en el análisis y comprensión de las herramientas seleccionadas para la sonificación de datos. Aquí presentamos un estudio de caso detallado para cada una de estas herramientas, proporcionando una visión práctica y detallada de su aplicación en este campo.

La inclusión de los casos de uso es esencial, ya que proporciona una comprensión completa y profunda de cómo estas herramientas se aplican en la práctica. Esto establece las bases para una evalua-

ción y una comparación significativa de sus capacidades y limitaciones en el contexto especializado de la sonificación de datos. Además, en nuestro enfoque multidisciplinario, analizamos las estadísticas de contribuciones de la comunidad en GitHub para cada herramienta, lo que arroja luz sobre el nivel de apoyo y el grado de desarrollo colaborativo impulsado por la comunidad de usuarios y desarrolladores en constante evolución.

4.1. Caso de uso

A continuación, se presenta un ejemplo práctico de cómo utilizar las herramientas seleccionadas. En este caso, se analizó minuciosamente la sintaxis y la estructura utilizadas para generar los sonidos a partir de una base de datos específica.

Para llevar a cabo las pruebas, se utilizó la base de datos denominada «informacion_lotes.csv», que contiene información detallada sobre la construcción de lotes en la ciudad de Medellín durante 2019. Esta base de datos proporciona los datos necesarios para generar el sonido inicial en las plataformas.

Este ejemplo práctico permite demostrar el potencial y la versatilidad de las herramientas investigadas en la creación y manipulación eficiente y efectiva de sonidos. Al utilizar la sintaxis y estructura adecuadas, se logró generar el primer sonido de manera simple y sin personalizaciones profundas. Esta acción resalta la capacidad de tales herramientas para realizar tareas básicas y sentar las bases para desarrollos más complejos en el ámbito del sonido en las plataformas.

- TwoTone

Para utilizar esta herramienta de forma web, se puede acceder a través de las siguientes direcciones URL: <https://twotone.>

io/ o <https://twotone-midiout-beta.netlify.app/>. En caso de que se haya instalado localmente, se debe ejecutar el comando npm start y acceder a localhost:9000 en el navegador.

Una vez adentro, es necesario hacer clic en «Get started» y seleccionar la base de datos informacion_lotes.csv (Figura 1). Seguidamente se pueden elegir las columnas que se desean sonificar y el sonido correspondiente para cada una de ellas. Además, es posible modificar parámetros como el volumen, la altura tonal (la afinación), la repetición del sonido y la velocidad en la que se leen los datos.

Es importante destacar que la herramienta cuenta con conexión MIDI, lo que permite enlazarla con sintetizadores, controladores y otros dispositivos que utilicen este protocolo. De este modo, tal acción ofrece la posibilidad de generar interacciones y una interpretación de los datos más completa. También es posible crear filtros mediante la comparación de las columnas seleccionadas con otras de la misma base de datos, para obtener resultados más precisos y ajustados a las necesidades del usuario.

La figura 1 muestra una vista general de la herramienta y sus opciones de configuración.

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/twotone>



Figura 1: visualización de TwoTone con la base de datos elegida para la comparación.
Nota: la imagen incluye algunos de los instrumentos y la cantidad de datos utilizados.

Fuente: elaboración de los autores.

• SuperCollider

SuperCollider es una plataforma que cuenta con una *library* (biblioteca) que permite leer directamente la base de datos. Sin embargo, para recrear un proceso similar al de TwoTone, que implica cargar un audio y modificar la velocidad de lectura de los datos, se implementó un *script* en Python. Este *script* fue diseñado para leer la base de datos y enviarlos de las columnas seleccionadas a SuperCollider mediante el protocolo OSC (Open Sound Control).

En virtud de su flexibilidad y eficiencia, el protocolo OSC permitió enviar los datos con una velocidad personalizada hacia SuperCollider, donde se realizaron las modificaciones necesarias para generar el sonido deseado. Esta combinación de herramientas y protocolos resultó efectiva para lograr el objetivo de modificar, en tiempo real, el sonido basado en los datos de la base de datos.

El código de Python que envía los datos a SuperCollider es el siguiente:

```
import csv
import time
import numpy as np
from pythonosc import osc_message_builder
from pythonosc import udp_client

def csv_generator(filename):
    with open(filename) as csvfile:
        reader = csv.reader(csvfile, delimiter=";")
        next(reader) # Omitir la primera fila (encabezados)
        print(reader)
        for row in reader:
            print(row)
            print(row[0])
            # Convertir los valores de la fila a los tipos de
            # datos adecuados
            codComuna = np.float64(row[1])
```

```

AreaLote = np.float64(row[3])
AreaCons = np.float64(row[4])
Zona = np.float64(row[5])
numPisos = np.float64(row[7])
yield (codComuna, AreaLote, AreaCons, Zona,
numPisos)
def send_messages(messages, client, address, time_delay):
for message in messages:
msg = osc_message_builder.OscMessageBuilder(address=address)
msg.add_arg(message[0], 'f')
msg.add_arg(message[1], 'f')
msg.add_arg(message[2], 'f')
msg.add_arg(message[3], 'f')
msg.add_arg(message[4], 'f')
msg = msg.build()

client.send(msg)

time.sleep(time_delay)
print(time_delay)

if __name__ == '__main__':
ip = "127.0.0.1"
port = 8888
client = udp_client.SimpleUDPClient(ip, port)

filename = "informacion_lotes.csv"
generator = csv_generator(filename)

address = "/csvSynth"
time_delay = 0.175 # tiempo de espera en segundos entre cada mensaje similar a TwoTone

# Enviar los mensajes en lotes para reducir la cantidad de tiempo de espera
batch_size = 100
messages = [message for message in generator]
for i in range(0, len(messages), batch_size):
send_messages(messages[i:i+batch_size], client, address, time_delay)
print(messages[i:i+batch_size], client, address, time_delay)

```

Una vez ejecutado este script, se procede a abrir SuperCollider para manipular el audio.

```

// Crear un servidor OSC en el puerto 57120
s.boot;

// Crear un cliente de OSC
n = NetAddr.new("127.0.0.1", 57120); // Reemplaza la dirección IP por la de tu computadora si es necesario
(
~codComuna = 0;
~areaLote = 0;
~areaCons = 0;
~zona = 0;
~numPisos = 0;
)

// Crear una función para procesar los mensajes OSC recibidos
(
OSCFunc({ |msg|
postln(msg[0].asString=="/csvSynth");
if(msg[0].asString == "/csvSynth") {
~codComuna = msg[1].asFloat;
~areaLote = msg[2].asFloat;
~areaCons = msg[3].asFloat;
~zona = msg[4].asFloat;
~numPisos = msg[5].asFloat;
}

// Imprimir los valores recibidos
postln("codComuna:" + ~codComuna);
postln("AreaLote:" + ~areaLote);
postln("AreaCons:" + ~areaCons);
postln("Zona:" + ~zona);
postln("numPisos:" + ~numPisos);
}
}, '/csvSynth').add;
)
(
SynthDef("sinteguan",{|codComuna, areaLote, areaCons, zona, numPisos|
var onda, envo, onda2, onda3, filtro;
onda = Saw.ar(zona);

```

```
onda2 = SinOsc.ar(areaCons);
onda3 = SinOsc.ar(areaLote);
filtro = LPF.ar(onda*onda2*onda3,numPisos*300+200);
envo = EnvGen.kr(Env.perc(0.05,0.8),1,doneAction:2);
Out.ar(0,filtro*envo);
}).add
)
(
Tdef(\sabor,{

inf.do{

Synth(\sinteguan,[\zona,\~zona,\areaCons,\~areaCons.midicps,\areaLote,\~areaLote.midicps,\numPisos,\~numPisos]);
0.25.wait;
}

});
)
Tdef(\sabor).play
```

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/supercollider>

- **Sonic Visualiser**

Para generar el sonido más básico utilizando este programa en Ubuntu, primero debe abrirse desde el panel de aplicaciones. Una vez abierto, se selecciona la opción «Convertir audio a partir de datos» en el menú «Archivo». A continuación, se elige la base de datos que se desea utilizar para sonorizar. Automáticamente se abrirá una ventana donde se pueden seleccionar las columnas específicas que se vayan a emplear (las mismas usadas en TwoTone y SuperCollider). Por último, solo es necesario hacer clic en el botón «Reproducir» para escuchar el sonido resultante (Figura 2).

El resultado puede escucharse en el siguiente enlace: <https://soundcloud.com/acorde-semillero/sonicVisualiser>

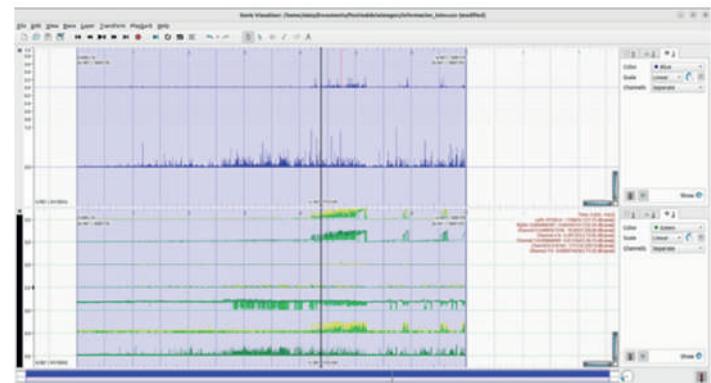


Figura 2: visualización de Sonic Visualiser con la base de datos elegida para la comparación
Nota: la imagen muestra las frecuencias que el programa eligió al convertir los datos.

Fuente: elaboración de los autores.

4.2. Discusión

4.2.1. Comparación entre TwoTone y SuperCollider

En términos de instalación, TwoTone es más fácil de instalar y configurar. Por otro lado, la instalación de SuperCollider puede ser más complicada, ya que requiere de una comprensión básica de programación y de los procesos de compilación. Sin embargo, una vez configurada, SuperCollider es una herramienta muy poderosa y flexible para la síntesis de sonido y el procesamiento de señales de audio.

En relación con la modificación, ambos softwares son altamente personalizables, pero SuperCollider es mucho más versátil y ofrece una amplia variedad de herramientas para la programación en tiempo real.

En términos de integración con Arduino y otros elementos externos, ambos softwares ofrecen una amplia variedad de herramientas para la comunicación con hardware externo. No obstante, la integración de SuperCollider con Arduino es mucho más robusta, ya que TwoTone solo

puede integrarse por medio del protocolo de comunicación MIDI.

4.2.2. Comparación entre TwoTone y Sonic Visualiser

En términos de instalación, Sonic Visualiser es más fácil de instalar y configurar. Sin embargo, su uso se limita principalmente a la visualización y el análisis de archivos de audio, lo que lo hace menos versátil que TwoTone.

En relación con la modificación, Sonic Visualiser es bastante limitado en términos de personalización, ya que no permite cambiar de instrumentos en las diferentes columnas de las bases de datos.

En términos de integración con Arduino y otros elementos externos, Sonic Visualiser no ofrece ninguna opción de integración, mientras que TwoTone brinda algunas opciones de integración básicas como MIDI.

4.2.3. Comparación entre SuperCollider y Sonic Visualiser

En términos de instalación, SuperCollider puede ser más complicado de instalar que Sonic Visualiser.

En relación con la modificación, SuperCollider es mucho más personalizable que Sonic Visualiser, ya que ofrece una amplia variedad de herramientas para la programación en tiempo real y la síntesis de sonido.

En términos de integración con Arduino y otros elementos externos, SuperCollider es mucho más robusto y ofrece más opciones de personalización que Sonic Visualiser.

5. Conclusiones

Elección de la herramienta de *software*: la elección de la herramienta de *software* para la sonificación es crucial y debe basarse en el nivel de conocimiento del usuario en programación, datos y música. Herramientas como TwoTone y Sonic Visualiser son ideales para aquellos con un conocimiento técnico limitado, ya que ofrecen interfaces gráficas intuitivas que no requieren una programación extensa.

TwoTone y Sonic Visualiser para usuarios con conocimiento limitado: estas herramientas son especialmente adecuadas para principiantes y personas sin experiencia previa en programación o síntesis de sonido. TwoTone se destaca por su accesibilidad en la conversión de datos en sonidos, mientras que Sonic Visualiser facilita la visualización y el análisis de datos de audio, lo que las hace aptas incluso para usuarios no técnicos.

SuperCollider para usuarios con conocimientos avanzados en programación: SuperCollider es una herramienta poderosa que permite una programación en tiempo real y una personalización avanzada de sonidos. Sin embargo, su uso eficaz requiere un nivel medio o alto de conocimiento en programación. Es la elección preferida para quienes buscan un control total sobre la síntesis de sonido y la manipulación de datos.

Personalización de sonidos: tanto SuperCollider como TwoTone lideran en la personalización de sonidos. SuperCollider brinda un control detallado sobre la síntesis y el procesamiento de señales, ideal para usuarios que desean crear sonidos únicos y experimentar con algoritmos de sonido personalizados. TwoTone, aunque más sencillo, ofrece una interfaz gráfica fácil de usar, pero con menos opciones de personalización en comparación con SuperCollider.

Limitaciones de las herramientas: cada herramienta tiene sus propias limitaciones. TwoTone se enfoca en la creación de sonidos a partir de datos, pero puede tener limitaciones en la síntesis de sonido más avanzada. SuperCollider, por otro lado, puede verse limitado por la habilidad de programación del usuario y los recursos de hardware disponibles, ya que puede ser intensivo en CPU.

Posibilidades creativas: SuperCollider es conocido por su versatilidad y puede utilizarse para crear una amplia variedad de sonidos, desde oscilaciones simples hasta texturas sonoras complejas. Esto lo hace ideal para proyectos artísticos y experimentales. TwoTone, aunque más sencillo, permite a usuarios con diversos niveles de conocimiento musical crear composiciones de manera intuitiva. Sonic Visualiser, en contraste, está diseñado principalmente para análisis científico y carece de las capacidades artísticas complejas de las otras herramientas.

Percepción auditiva y resultados artísticos: los resultados de Sonic Visualiser tienden a ser más científicos y analíticos, lo que puede resultar en la falta de elementos artísticos, como sonidos melódicos o de larga duración. SuperCollider destaca en términos de creatividad, pero requiere un conocimiento previo para lograr resultados artísticos complejos. TwoTone se encuentra en un punto intermedio, ofreciendo una forma accesible de crear música a partir de datos, pero con ciertas limitaciones en la complejidad musical.

Futuros trabajos: se sugiere la exploración de aspectos estéticos y filosóficos en la sonificación de datos para la creación de obras sonoras más reflexivas y profundas en el futuro. Además, la colaboración interdisciplinaria y la investigación de técnicas avanzadas pueden enriquecer la representación de datos y mejorar la calidad artística de las obras sonoras resultantes.

5.1. Limitaciones

La investigación se centró en generar el sonido más sencillo y básico de cada plataforma, evitando entrar en discusiones filosóficas sobre su creación o en reflexiones estéticas. Así, se buscó brindar un primer acercamiento sonoro en cada plataforma y comparar las dificultades encontradas en el proceso.

Como resultado, es importante destacar que los sonoros pueden ser notablemente monótonos y planos si se comparan con otros géneros musicales más complejos y elaborados en términos de arreglos musicales, instrumentación y estructuras melódicas. No obstante, también es cierto que algunos artistas y compositores de música experimental pueden buscar intencionalmente una estética similar a la de los sonoros, explorando la simplicidad y la repetición como elementos expresivos en su música.

En esta investigación, el enfoque se centró en generar los sonidos más simples posibles utilizando los recursos disponibles en cada plataforma. Esto implicó la utilización de osciladores fundamentales, formas de onda básicas y tonos puros, sin agregar elementos adicionales como modulaciones, efectos o procesamiento sonoro complejo. Se abordó cada plataforma desde una perspectiva técnica y objetiva, evitando cualquier interpretación subjetiva o intenciones artísticas. El propósito principal era explorar las capacidades más básicas en la generación de sonido de cada plataforma y evaluar las limitaciones y desafíos encontrados en el proceso. Sin embargo, es esencial recordar que la elección de la herramienta de *software* para la sonificación de datos puede depender en gran medida del nivel de conocimiento del usuario en áreas como la programación, la mu-

sica y la gestión de datos. Estos factores influyen en la capacidad del usuario para aprovechar al máximo las herramientas disponibles y personalizar los resultados según sus necesidades específicas.

En muchas plataformas, la generación de un sonido básico resultó más complicada de lo esperado, debido a limitaciones técnicas, falta de opciones directas para generar ondas puras o una interfaz poco intuitiva. Además, algunos sistemas presentaron una calidad sonora inferior o

una respuesta limitada en relación con la variación sonora.

Es importante resaltar que estos resultados no deben ser considerados como un juicio absoluto sobre las capacidades de cada plataforma. El objetivo de este estudio fue puramente investigativo y no se tuvieron en cuenta otros aspectos que podrían influir en la percepción de la calidad sonora como la capacidad de síntesis, la complejidad de los algoritmos o las opciones de personalización.

Referencias

- Aaron, S. «Sonic Pi-The live coding music synth for everyone». Sonic Pi, 2018. <https://sonic-pi.net/>
- Barsetti-Nerland, Daniel Edward. «Creating a responsive visualization that reacts with music in real time: integrating Ableton Live 9 and Cycling'74 max for live into a musical performance». Tesis de pregrado, University of Arizona, Tucson, 2016. <https://repository.arizona.edu/handle/10150/612546>
- Cannam, Chris, Christian Landone, Mark Sandler y Juan Pablo Bello. «The sonic visualiser: a visualisation platform for semantic descriptors from musical signals». *ISMIR 2006 7th International Conference on Music Information Retrieval*, 324–327. Victoria, BC, Canadá: 2006. <https://nyuscholars.nyu.edu/en/publications/the-sonic-visualiser-a-visualisation-platform-for-semantic-descri>
- Castillo Vázquez, Alina, Diedmar Céspedes Cabrera, Mercedes Jiménez Betancourt y Gustavo Rodríguez Rosabal. «Alternativa didáctica para el estudio de los órganos sensitivos en la enseñanza de la Biología 3». *Revista Dilemas Contemporáneos: Educación, Política y Valores*, vol. 8, n.º 1 (2020). <https://dilemascontemporaneoseduacionpoliticaayvalores.com/index.php/dilemas/article/view/2386>
- De la Vega, Gonzalo, Leonardo Martín Exequiel Domínguez, Johanna Casado, Beatriz García. «SonoUno Web: An innovative user centred web interface». Arxiv (2022). <https://arxiv.org/pdf/2302.00081.pdf>
- Gárate Piñones, I. A. J. «Doroti». Tesis doctoral, Universidad Andrés Bello, 2020.
- Garrett, Richard. «Audio Spray Gun 0.8-The generation of large sound-groups and their use in three-dimensional spatialisation». *ICMC* (2015). <https://quod.lib.umich.edu/cgi/p/pod/dod-idx/audio-spray-gun-08-the-generation-of-large-sound-groups.pdf?c=icmc;idno=bbp2372.2015.071;format=pdf>
- Martínez Cantón, Carlos. «Diseño y construcción de un detector de metales basado en un microcontrolador». Tesis de pregrado, Universitat Politècnica de Catalunya, España, 2016. <https://upcommons.upc.edu/handle/2117/98926?locale-attribute=es>
- Peralta Rocha, Bryan Marcelo. «Inspección y pintura del avión Escuela Hawker Siddeley 125-400, mediante información y datos técnicos para la unidad de gestión de tecnologías UGT-ESPE». Tesis de pregrado, Universidad de las Fuerzas Armadas, Sangolquí, Ecuador, 2018. <https://repositorio.espe.edu.ec/bitstream/21000/27745/1/M-ESPEL-CMA-0478.pdf>
- Quimica.es. «Estetoscopio». Quimica.es. <https://www.quimica.es/enciclopedia/Estetoscopio.html>

- Ramírez Arias, Leonardo y Julián Uribe Cortés. «El contador Geiger como herramienta en la enseñanza del concepto de radiación». *Góndola, Enseñanza y Aprendizaje de las Ciencias*, vol. 2, n.º 1 (2007): 33-36. <https://revistas.udistrital.edu.co/index.php/GDLA/article/download/5304/6935/0>
- Remiro Gargallo, Andrea Vicenta. «Técnicas numéricas aplicadas a la propagación de sonido en el medio marino». Tesis doctoral, Universitat Politècnica de València, España, 2019. <http://hdl.handle.net/10251/128636>
- Rodríguez Mateos, David y Alicia Tapia López. «Sonificación y periodismo: la representación de datos mediante sonidos». *Revista de Comunicación*, vol. 22, n.º 1 (2023): 413-433. <http://dx.doi.org/10.26441/rc22.1-2023-3022>
- Ruiz-del-Olmo, Francisco Javier y José Vertedor. «Procesos, herramientas y prácticas de la sonificación». Málaga, España, Universidad de Málaga, 2016. <http://hdl.handle.net/10630/12429>
- Sánchez Corzo, Andrea del Pilar. «Sonorización de señales de EEG basada en estructuras musicales». Tesis de pregrado, Universidad de los Andes, Bogotá, 2012. <https://repositorio.uniandes.edu.co/bitstream/handle/1992/25182/u627972.pdf?sequence=1&isAllowed=y>
- Saraví, Fernando. «Laennec, el método anatomooclínico y la invención del estetoscopio. Parte 1: De Bretaña a París, hasta la gran invención». *Revista Médica Universitaria*, vol. 8, n.º 2 (2012). https://bdigital.unc.edu.ar/objetos_digitales/4769/rmuhistoriasdelamedicina.pdf
- Schaeffer, Pierre, Christine North y John Dack. *Treatise on Musical Objects: An Essay Across Disciplines*. California: University of California Press, 2017. <https://www.jstor.org/stable/10.1525/j.ctt1qv5pqb>
- Staples, Anne. «El abuso de las campanas en el siglo pasado». *Historia Mexicana*, vol. 27, n.º 2 (1977): 177-194. <https://www.jstor.org/stable/25135599>
- Vercoe, B., J. Fitch, J. Piche, P. Nix, R. Boulan-ger, R., Ekman y A. Cabrea. «The canonical Csound reference manual». Csound, 2007. <https://csound.com/docs/manual/index.html>
- Vickers, Paul y Bennet Hogg. «Sonification abstraite / Sonification concrète: An aesthetic perspective space for classifying auditory displays in the Ars Musica domain». En *Proceedings of the 12th International Conference on Auditory Display* (210-216). Londres, 2006. <https://typeset.io/pdf/sonification-abstraite-sonification-concrete-an-aesthetic-1urtp0ql.pdf>
- Walker, Bruce N. y Joshua T. Cothran. «Sonification sandbox: a graphical toolkit for auditory graphs». *Proceedings of the 2003 International Conference on Auditory Display*. Atlanta, GA: Georgia Institute of Technology, 2003.
- Wang, Ge, Dan Trueman, Scott Smallwood y Perry R. Cook. «The laptop orchestra as classroom». *Computer Music Journal*, vol. 32, n.º 1 (2008): 26-37. <https://www.jstor.org/stable/40072662>
- Wang, Hengbo. «Dear diary: an investigation of emotionally engaging computer-game narratives». Tesis de maestría, Auckland University of Technology, Australia, 2015. <http://hdl.handle.net/10292/9137>
- Wikipedia. «Sonar». Wikipedia, 21 de marzo de 2019. <https://es.wikipedia.org/wiki/Sonar>
- Wilson, Scott, David Cottle y Nick Collins. *The SuperCollider Book*. Cambridge, MA: The MIT Press, 2011.