

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

**“MODELAMIENTO DEL RUIDO AMBIENTAL EN ZONAS URBANAS Y
SECTORIZACIÓN DE LOS PUNTOS CRÍTICOS EN LA CIUDAD DE
GUAYAQUIL”**

EXAMEN COMPLEXIVO

Previa la Obtención del Título de:

MAGISTER EN TELECOMUNICACIONES

AUTOR:

DANIEL FERNANDO MACIAS AGUILAR

GUAYAQUIL – ECUADOR

Año: 2025

DEDICATORIA

"A mis padres, el Master Freddy Macías y la Licenciada Jackeline Aguilar, por ser mi inspiración y mi orgullo. Los amo y admiro más que palabras pueden expresar. Mi sueño de niño siempre fue llenarlos de orgullo, y hoy puedo decir que lo he logrado gracias a su amor y apoyo.

A mi novia, la Psicóloga Karla Sinchiguano, por ser mi roca en momentos difíciles. Su amor y apoyo incondicional me han permitido crecer y alcanzar mis metas.

A mis hermanos y mejores amigos, el Master Freddy Macías y la Master Chrystel Macías, los amo y los aprecio más de lo que pueden imaginar.

A mi abuelo Jacinto Aguilar que siempre será para mí uno de los mejores ejemplos de valentía y fuerza ,
A mi abuelo Fernando Macías y su frase que me acompaña siempre "El que pesa por quintales no se fija en media libra". A mi prima Jessenia Andrade y mi primo Alberto Cedeño, que nunca se rindan en su camino. Dios los bendiga.

Estoy agradecido con la familia que Dios me ha dado. Sin ellos, nada de esto hubiera sido posible.

"Y recuerda, nadie podrá hacerte frente en todos los días de tu vida, siempre estaré contigo. Sé fuerte y valiente".
Josué 1:5-6.

Daniel Macías.

AGRADECIMIENTOS

Agradezco sinceramente a Dios, por su guía y bendición constante en mi vida, jamás me dejó solo, él siempre estuvo ahí.

A mi papá, el Master Freddy Macías, y a mi mamá, la Licenciada Jackeline Aguilar, por su amor incondicional y dedicación incansable. Han sido mi roca, mi guía y mi inspiración. Su sabiduría y experiencia me han brindado la orientación necesaria para superar cualquier obstáculo y alcanzar mis metas. Su amor y presencia constante en mi vida me han dado la fuerza y la confianza para seguir adelante. Les agradezco por ser mis modelos para seguir, por enseñarme los valores y principios que me han llevado a donde estoy hoy.

A mi novia, la Psicóloga Karla Sinchiguano, por su apoyo y comprensión. Su presencia en mi vida ha sido un motivador constante, tu amor y dedicación significan el mundo para mí.

A mis hermanos, Freddy Macías y Chrystel Macías, por ser más que solo hermanos, son mis mejores amigos. Siempre pueden contar conmigo, en las buenas y en las malas, espero que sepan que los amo.

A todos ellos, mi más sincero agradecimiento por haber sido parte de este proceso y haberme permitido alcanzar este logro.

Daniel Macías.

DECLARACIÓN EXPRESA

Yo Daniel Fernando Macias Aguilar acuerdo y reconozco que: La titularidad de los derechos patrimoniales de autor del proyecto de graduación corresponderá al autor, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores. El o los estudiantes deberán procurar en cualquier caso de cesión de sus derechos patrimoniales incluir una cláusula en la cesión que proteja la vigencia de la licencia aquí concedida a la ESPOL.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, secreto empresarial, derechos patrimoniales de autor sobre software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por mí/nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que me corresponda de los beneficios económicos que la ESPOL reciba por la explotación de mi innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique al autor que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 29 de enero del 2025.

Daniel Fernando Macias Aguilar

Evaluadores

Ing. María Antonieta Álvarez, PhD.

Evaluador

Ing. Eduardo Chancay, Mgtr.

Evaluador

RESUMEN

El ruido sonoro es un problema ambiental creciente que afecta tanto a la calidad de vida humana como a los ecosistemas naturales. En áreas urbanas e industriales, la exposición prolongada a niveles elevados de ruido puede generar estrés, problemas cardiovasculares y trastornos del sueño en las personas. En el entorno silvestre, el ruido altera los patrones de comunicación, reproducción y comportamiento de muchas especies, poniendo en riesgo la biodiversidad.

Para abordar esta problemática, se propone realizar una medición sistemática del ruido utilizando herramientas de Internet de las Cosas (IoT) y técnicas de Machine Learning (ML). Los dispositivos IoT permitirán recopilar datos en tiempo real sobre la intensidad, frecuencia y duración del ruido en diferentes ubicaciones, creando un mapa sonoro dinámico. Estos dispositivos, equipados con sensores acústicos, se distribuirán estratégicamente para garantizar una cobertura representativa.

Con los datos recopilados, se aplicarán modelos de Machine Learning para analizar patrones y tendencias en la generación del ruido. Esto incluirá la identificación de fuentes principales, horarios críticos y la evaluación de su impacto en áreas urbanas y naturales. Los modelos predictivos permitirán anticipar escenarios futuros y simular la efectividad de posibles soluciones.

Finalmente, se desarrollarán técnicas de mitigación basadas en los resultados del análisis. Estas podrían incluir barreras acústicas inteligentes, ajustes en la planificación urbana y estrategias para minimizar la emisión de ruido en zonas críticas.

Palabras Clave: Contaminación auditiva, Ruido urbano, Aprendizaje de Maquina, Internet de las Cosas,

ABSTRACT

Sound noise is an increasing environmental problem that affects both the quality of human life and natural ecosystems. In urban and industrial areas, prolonged exposure to high noise levels can cause stress, cardiovascular issues, and sleep disorders in humans. In the wild, noise disrupts communication, reproduction, and behavioral patterns of many species, putting biodiversity at risk.

To address this issue, a systematic noise measurement is proposed using Internet of Things (IoT) tools and Machine Learning (ML) techniques. IoT devices will enable real-time data collection on the intensity, frequency, and duration of noise across various locations, creating a dynamic sound map. These devices, equipped with acoustic sensors, will be strategically distributed to ensure representative coverage.

With the collected data, Machine Learning models will be applied to analyze patterns and trends in noise generation. This will include identifying primary sources, critical timeframes, and assessing the impact on urban and natural areas. Predictive models will allow for anticipating future scenarios and simulating the effectiveness of potential solutions.

Finally, mitigation techniques will be developed based on the analysis results. These could include intelligent acoustic barriers, adjustments in urban planning, and strategies to minimize noise emissions in critical areas. This approach combines advanced technology with a sustainable vision, aiming to reduce the negative effects of noise on human health and ecosystems.

The use of IoT and ML not only enhances the precision and efficiency of noise measurement but also facilitates the implementation of preventive and corrective measures, fostering a healthier and more balanced environment.

Keywords: *Noise pollution, Urban noise, Machine Learning, Internet of Things*

INDICE GENERAL

ABREVIATURAS	10
ÍNDICE DE FIGURAS	10
ÍNDICE DE TABLAS	10
1. INTRODUCCIÓN	11
1.1 ANTECEDENTES.....	13
1.2 DESCRIPCIÓN DEL PROBLEMA	15
1.3 JUSTIFICACIÓN DE LA PROPUESTA	17
1.4 DELIMITACIÓN	17
1.4.1 DELIMITACIÓN TEMPORAL	17
1.4.2 DELIMITACIÓN ESPACIAL.....	18
1.5 OBJETIVOS.....	18
1.5.1 OBJETIVO GENERAL	18
1.5.2 OBJETIVOS ESPECÍFICOS.....	18
1.6 MARCO TEÓRICO	19
1.6.1 ESPECTRO AUDIBLE	19
1.6.2 MARCO TEORICO	21
1.6.2.1 MACHINE LEARNING EN EDGE COMPUTING	21
1.6.2.2 DESCRIPCIÓN DE LOS MODELOS DE DATOS UTILIZADOS.....	21
1.6.2.3 REDES NEURONALES DENSAS (DNN).....	22
1.6.2.4 REDES NEURONALES CONVOLUCIONALES (CNN).....	24
1.6.2.5 EDGE IMPULSE.....	24
2. DISEÑO	25
2.1.1 INTRODUCCIÓN	25
2.1.2 CONTEXTO Y RELEVANCIA	26
2.1.3 ESTADO DEL ARTE EN SISTEMAS DE PREDICCIÓN AMBIENTAL	26
2.1.4 CARACTERÍSTICAS ESPECÍFICAS IMPLEMENTADAS	27
2.1.5 DETALLES DE LA INFRAESTRUCTURA DE EQUIPOS.....	27
2.2 ELECCIÓN DE LOS MODELOS Y LA PLATAFORMA.....	30
3. ANÁLISIS DE RESULTADOS	33
3.1 ANÁLISIS PRELIMINAR DE LOS DATOS	33
3.2 ANÁLISIS DE LA DISTRIBUCIÓN.....	36
3.3 ANÁLISIS DE RENDIMIENTO DEL MODELO DE MACHINE LEARNING	38

3.4	RESULTADOS DEL MODELO DE CLASIFICACIÓN.....	40
4.	CONCLUSIONES Y RECOMENDACIONES.....	41
4.1	CONCLUSIONES.....	41
4.2	RECOMENDACIONES.....	42
5.	BIBLIOGRAFÍA.....	42
6.	APÉNDICES	45

ABREVIATURAS

SVM	Máquina de Soporte Vectorial
RNN	Red Neuronal Replicadora
CNN	Red Neuronal Convolutiva
DNN	Red Neuronal Profunda
ML	Aprendizaje de Máquina
IoT	Internet de las Cosas
ROC	Curva Característica Operativa del Receptor
EPA	Agencia de Protección Ambiental
IAQ	Calidad del Aire en Ambiente Interior
FTT	Transformada Rápida de Fourier
ANSI	Instituto Nacional Americano de Estándares
MEMS	Sistemas Microelectromecánicos
RBM	Máquina de Boltzmann Restringida
GMM	Modelo Gaussiano Mixto
PCA	Análisis de Componentes Principales

ÍNDICE DE FIGURAS

Figura 1 Patrón del ruido cerca de la zona de medición	11
Figura 2 Ubicación de la zona de medición.....	18
Figura 3 Transformación del audio capturado	21
Figura 4 Prototipo de Recolección de Datos.....	27
Figura 5 Resumen Estadístico.....	33
Figura 6 Diagrama de Dispersión Temperatura vs Humedad.....	34
Figura 7 Diagrama de Dispersión Presencia de Ruido vs Hora.....	36
Figura 8 Diagrama de Barra del Nivel de Sonido	37
Figura 9 Matriz de Correlación	38
Figura 10 Resultados del Entrenamiento.....	39
Figura 11 Resultado con el conjunto de datos de prueba.....	40

ÍNDICE DE TABLAS

Tabla 1 Arquitectura CNN implementada	24
Tabla 2 característica de Edge Impulse.....	25
Tabla 3 Características Específicas	27
Tabla 4 Infraestructura Usada	28

1. INTRODUCCIÓN

El ruido ambiental influye en la calidad de vida. Los coches, los aviones y el bullicio comercial pueden perturbar el sueño, crear estrés o problemas de audición e influir en el desarrollo cognitivo de los niños [1]. Otro aspecto del ruido ambiental es que está directamente relacionado con las actividades y estilos de vida humanos. Como se muestra en la **Figura 1**, un patrón de nivel de ruido diario monitoreado cerca de una carretera revela la asociación entre el ruido ambiental y la actividad humana. Desde la medianoche hasta la madrugada, que corresponde al tiempo de sueño de muchas familias, el nivel de ruido promedio es bajo en comparación con el resto del día. Después de eso, el nivel de ruido promedio aumenta debido a la necesidad de viajar entre la casa y la escuela o los lugares de trabajo. Las variaciones en el nivel de ruido son significativas entre el día y la noche. Por lo tanto, el monitoreo y análisis del ruido ambiental ha sido un área de investigación activa durante décadas [2].

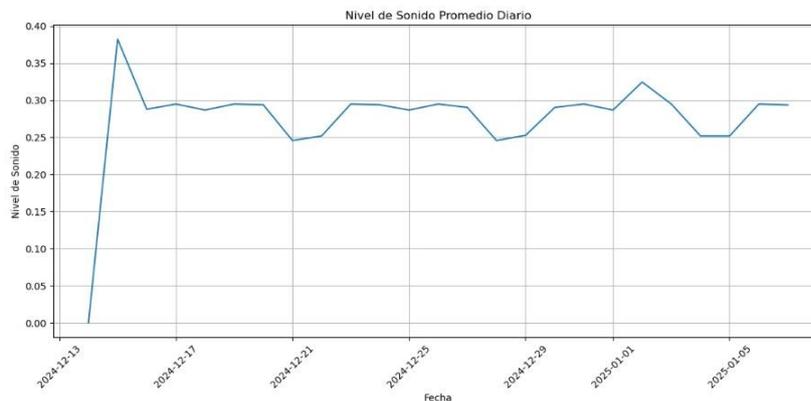


Figura 1 Promedio del ruido en el lugar de medición

Una herramienta importante para el análisis del ruido ambiental es la detección de anomalías o valores atípicos, en la que se define un proceso generativo para el patrón de ruido diario a fin de proporcionar las características típicas del ruido de un área en particular, y las anomalías son puntos o eventos que es poco probable

que genere el modelo generativo [3]. Dado que el monitoreo a largo plazo siempre produce una cantidad masiva de datos, los investigadores y analistas no pueden investigar cada punto de datos. Por lo tanto, las anomalías marcan eventos e instancias potencialmente interesantes que merecen una investigación más profunda [4]. Como ejemplos, las anomalías en calles residenciales pueden incluir sirenas de policía o ambulancias, accidentes automovilísticos o personas gritando y durante los apagones el ruido de los generadores eléctricos. Si también se instalan cámaras de vigilancia, las anomalías en el ruido ambiental podrían activar selectivamente la grabación y transmisión de audio sin procesar y de la cámara, lo que reduciría el tráfico de red y el almacenamiento de datos [5].

La detección de anomalías en el ruido ambiental acústico se enfrenta a varios desafíos. En primer lugar, para proteger la privacidad del habla, el algoritmo solo se debe aplicar a mediciones generales, como la intensidad del ruido, la banda de octava o las bandas de tercio de octava en intervalos considerablemente mayores que la duración del fonema, pero no directamente al audio sin procesar [6]. En segundo lugar, las definiciones de normalidad y anomalía dependen del tiempo en un área monitoreada determinada, porque los patrones de nivel de ruido cambian con el tiempo, como se muestra en la **Figura 1**. En tercer lugar, si se utilizan bandas de octava o bandas de tercio de octava, el algoritmo tiene que funcionar con datos de alta dimensión [7].

Afortunadamente, la variación del ruido acústico está fuertemente relacionada con las actividades humanas; por lo tanto, es razonable suponer una periodicidad de 24 horas en los modelos generativos en un entorno urbano. La periodicidad sugiere medios para reducir la complejidad del algoritmo de detección de anomalías. Esta tesis aplica enfoques bien conocidos en la detección de

anomalías, incluyendo la máquina de vectores de soporte (SVM) de una clase, la red neuronal replicadora (RNN) y la detección de anomalías basada en el análisis de componentes principales a un monitor de intensidad de ruido de banda de octava continua en un área residencial, antes de proponer una detección de anomalías basada en el análisis de componentes principales que varía en el tiempo y que mejora el rendimiento significativamente [8]. El método propuesto trata las mediciones en cada hora de forma independiente. En cada hora, las mediciones típicas se generan aproximadamente mediante una distribución gaussiana multivariada, y las anomalías son muestras de entrada que es poco probable que estén presentes bajo la distribución normativa correspondiente.

1.1 ANTECEDENTES

Muchos trabajos en monitoreo de ruido ambiental recientemente se han enfocado en la detección y clasificación de eventos acústicos. Para ello utilizaron una red neuronal profunda convolucional que tiene “3 capas convolucionales intercaladas con 2 operaciones de agrupamiento, seguidas de 2 capas completamente conectadas (densas)”. En los pasos de preprocesamiento, se transforman datos de audio sin procesar en una representación de espectrograma MEL a escala logarítmica antes de extraer bloques de tiempo-frecuencia como entrada para las redes. La detección de gritos y disparos es otro tema de estudio, en donde se convirtieron cuadros de audio de 23 ms en vectores de características que incluyen tasa de cruce por cero (ZCR), coeficientes espectrales de frecuencia MEL (MFCC) y algunas otras mediciones basadas en el espectro y la distribución antes de usar dos modelos de mezcla gaussiana independientes para discriminar disparos y gritos del ruido ambiental respectivamente. Las factorizaciones matriciales como la factorización matricial no negativa (NMF), el análisis de

componentes principales (PCA) y sus variantes que buscan buenas representaciones de escenas acústicas ambientales son examinadas.

Además, se ha estudiado la detección de anomalías en eventos acústicos, suelen utilizar un modelo de mezcla gaussiana (GMM) para formar representaciones estadísticas de eventos normales, estableciendo un umbral de probabilidad de los datos entrantes en función de anomalías seleccionadas devueltas por el GMM, en la cual se aplica una Máquina de Boltzmann Restringida (RBM) y una RBM condicional como modelo generativo del entorno acústico, y las anomalías se identifican en función de su probabilidad.

En todos los trabajos antes mencionados, los eventos acústicos transitorios son sujetos de clasificación y detección. En otras palabras, la escena acústica siempre debe estar representada con suficiente información que pueda reconstruir el audio crudo correspondiente. Por lo tanto, estos enfoques pueden no ser fáciles de implementar en entornos públicos debido a preocupaciones de privacidad sobre el habla humana. Afortunadamente, el monitoreo del ruido ambiental por medio de mediciones del nivel de presión sonora, banda de octava, banda de tercio de octava se puede realizar en áreas públicas. El hardware y la infraestructura están disponibles para el monitoreo del ruido hasta a escala de ciudad con un costo razonable. Implementar una matriz de micrófonos de sistemas microelectromecánicos (MEMS) de bajo costo que cumple con el estándar IEC 61672-1 para medidores de nivel de sonido electroacústico. Además, de proporcionar una red de sensores diseñada para implementación a escala de ciudad. Por lo tanto, creo que, en muchos casos, la clasificación y detección aplicadas para el monitoreo ambiental deberían comenzar con el supuesto de que

solo se encuentran disponibles como entradas los niveles de presión sonora en intervalos de al menos un segundo.

Teniendo en cuenta este estudio de trabajos relacionados, nuestras contribuciones se pueden resumir de la siguiente manera: en primer lugar, el estudio se lleva a cabo bajo el supuesto de que solo se dispone de datos brutos del nivel de presión sonora. En segundo lugar, se ilustra la naturaleza no estacionaria de los niveles de sonido en áreas residenciales y sus patrones diarios. En tercer lugar, se informan diferentes algoritmos de detección de anomalías aplicados a un conjunto de datos de nivel de presión sonora de monitoreo continuo. Por último, la extensión de la detección robusta de anomalías mediante la introducción de parámetros de media y covarianza constantes por partes en la distribución gaussiana multivariante del modelo generativo produce una detección de anomalías que puede adaptarse a la dinámica del nivel de ruido residencial.

1.2 DESCRIPCIÓN DEL PROBLEMA

La problemática del ruido ambiental en las zonas urbanas de Guayaquil se ha intensificado en los últimos años debido al crecimiento poblacional, el aumento del parque automotor, la expansión de actividades comerciales y recientemente el ruido producido por los generadores eléctricos. Este fenómeno, que afecta de manera significativa la calidad de vida de los habitantes, se manifiesta en diversos entornos como las áreas residenciales y los sectores industriales.

El tráfico vehicular, una de las principales fuentes de contaminación acústica, genera niveles de ruido que exceden los límites establecidos por las normativas europeas Directive 2002/49/EC. A esto se suman el ruido de transporte público, las bocinas y las obras de construcción. En sectores con alta actividad comercial,

como los mercados o avenidas principales, los niveles de ruido son persistentes a lo largo del día, afectando no solo a los trabajadores y residentes, sino también a los animales.

El ruido ambiental tiene consecuencias directas en la salud pública. Estudios han vinculado la exposición prolongada a altos niveles de ruido con problemas como estrés, trastornos del sueño, dificultades de concentración, e incluso afecciones cardiovasculares. En Guayaquil, la falta de una gestión eficaz del ruido urbano ha generado un ambiente sonoro deteriorado, donde las zonas de mayor actividad carecen de una planificación adecuada para mitigar este tipo de contaminación.

A pesar de que existe normativa en la ciudad para regular los niveles de ruido, su implementación ha sido limitada. La falta de monitoreo constante y de sanciones efectivas a los infractores son factores que contribuyen al incremento del ruido ambiental. Además, los avances en tecnología para el control y monitoreo de este fenómeno, como el uso de sensores y dispositivos de medición en tiempo real, aún no han sido implementados de manera significativa en la ciudad, lo que impide un control más eficiente y una mejor respuesta ante los problemas acústicos.

El desafío en Guayaquil no solo radica en controlar el ruido actual, sino también en prevenir un incremento futuro mediante una adecuada planificación urbana, la creación de zonas de amortiguamiento acústico y la implementación de políticas públicas que integren el monitoreo continuo del ruido ambiental y la participación de la ciudadanía en su mitigación. Por lo cual un estudio para determinar el estado de los niveles de ruido es el primer paso para mitigar y gestionar de forma adecuada.

1.3 JUSTIFICACIÓN DE LA PROPUESTA

Tres pilares básicos sirvieron como base para la selección de piezas y configuraciones específicas para este sistema, cada una de las cuales fue creada para abordar aspectos cruciales de funcionalidad y rendimiento. El sistema ha mejorado en términos de precisión de las predicciones al desarrollar una sólida comprensión de series temporales complejas, respaldada por un control eficiente de una variedad de factores externos. También ha recibido mucha atención la habilidad del equipo para aclimatarse con éxito a los datos.

Otra base crucial es la eficiencia computacional, que está determinada por una optimización específica para dispositivos sofisticados y garantiza un uso limitado de los recursos sin comprometer el rendimiento. El sistema fue diseñado para garantizar un procesamiento efectivo en tiempo real y al mismo tiempo lograr un equilibrio entre el tiempo de respuesta y la eficiencia energética.

La capacidad de mantener y hacer crecer el sistema es el tercer elemento. Como resultado, se ha desarrollado un marco modular que facilita la integración de nuevos componentes en los sistemas actuales, así como la renovación de componentes individuales. Al utilizar un diseño modular, el sistema puede adaptarse a las necesidades cambiantes y avanzar sin requerir una revisión completa.

1.4 DELIMITACIÓN

1.4.1 DELIMITACIÓN TEMPORAL

El proyecto de tesis se realizará en un plazo de dos meses. Durante ese tiempo, todas las situaciones problemáticas se registrarán mediante mediciones la presión sonora para determinar posibles fuentes de ruido.

1.4.2 DELIMITACIÓN ESPACIAL

“La investigación se llevará a cabo en las cercanías del “Parque Histórico”, ubicada en la ciudadela Entre Ríos entre calle Esmeralda y avenida central, vía Samborondón en la ciudad de Guayaquil.

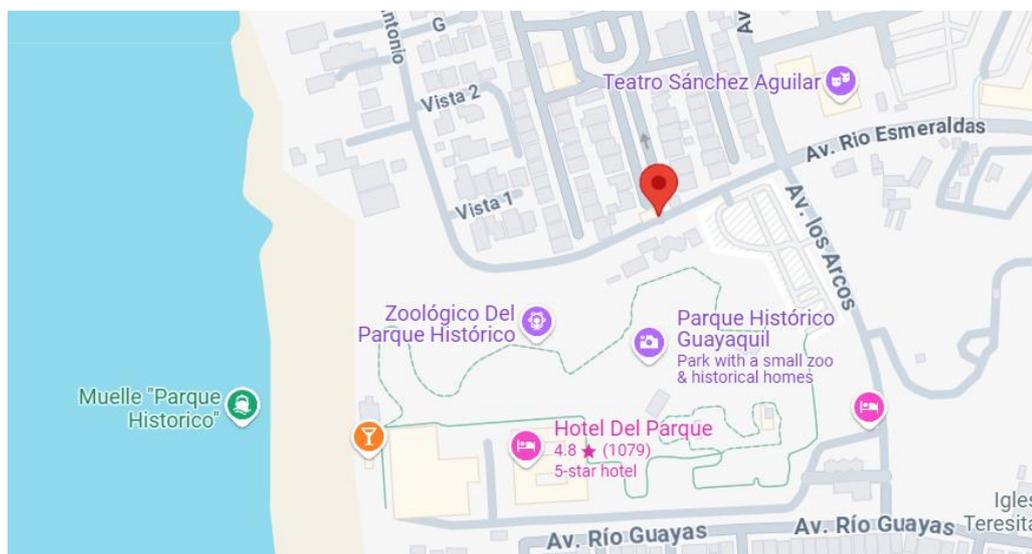


Figura 2 Ubicación de la zona de medición

1.5 OBJETIVOS

1.5.1 OBJETIVO GENERAL

Efectuar un análisis sonoro exhaustivo para reconocer las fuentes de ruido más significativas en la ciudad, valorar su influencia en la población y formular tácticas de mitigación que ayuden a potenciar la calidad del sonido del ambiente urbano.

1.5.2 OBJETIVOS ESPECÍFICOS

1. Diseñar un plan de medición para recopilar datos acústicos en diferentes zonas de la ciudad, utilizando equipos y sensores acústicos para identificar patrones de ruido, fuentes predominantes y horarios de mayor incidencia.

2. Desarrollar modelos matemáticos y de aprendizaje automático que permitan predecir eventos de ruido en función de los datos recolectados, considerando factores como horarios, condiciones climáticas, densidad vehicular y actividades humanas.
3. Desarrollar una plataforma tecnológica que incorpore el seguimiento en tiempo real de los niveles de ruido, emitiendo alertas automáticas si se superan los límites críticos, y brindando datos valiosos para la toma de decisiones y la planificación de la ciudad.

1.6 MARCO TEÓRICO

Este capítulo proporciona el marco teórico para la discusión y las explicaciones de los capítulos siguientes. Se introducen las medidas de banda audible, seguidas de una discusión de las propiedades estadísticas, incluidas las distribuciones normales, log-normal y chi-cuadrado, antes de considerar el análisis de componentes principales (PCA). A continuación, se estudian las técnicas de detección de anomalías que comprenden detecciones de anomalías basadas en PCA, máquinas de vectores de soporte (SVM) de una clase y redes neuronales replicadoras (RNN). Por último, se introducen las métricas de rendimiento de los algoritmos de detección de anomalías que consisten en recuperación, precisión y medida.

1.6.1 ESPECTRO AUDIBLE

Como se indica en la norma ANSI S1.43-1997 (r 2007), un medidor de nivel de presión sonora puede dividir el espectro de frecuencia en bandas de octava y proporcionar mediciones del nivel de energía promedio de cada banda en

intervalos de un segundo. Por lo tanto, la legislación estadounidense aprueba la recopilación de mediciones de bandas de octava en entornos públicos, ya que no se conoce ningún método para reconstruir la información del habla a partir de estas mediciones.

Además, un estudio científico intensivo del habla humana ha determinado que la información sobre las palabras pronunciadas y la identidad del hablante se transporta en la estructura espectro-temporal (tiempo-frecuencia) fina de la señal. Por lo tanto, si realizamos un muestreo insuficiente en este dominio (o en otro dominio del que se pueda demostrar que esta información es irrecuperable), el habla y el hablante son fundamentalmente irrecuperables.

Además, el nivel de ruido en bandas de octava proporciona información más rica que una única intensidad de ruido con la misma frecuencia de muestreo y ahorra recursos de procesamiento, ya que se puede implementar en hardware.

En esta tesis, se recopila el audio sin procesar para validar los resultados del análisis; por lo tanto, necesitamos transformar los datos de audio en características de banda de octava. Los datos de audio sin procesar se dividen en cuadros superpuestos. En cada cuadro, la energía de la señal se presenta en forma de componentes de banda de octava. Aplicando el teorema de Parseval y la teoría de procesamiento de señales, se puede calcular un componente de banda de octava sumando los cuadrados de las magnitudes de la transformada rápida de Fourier (FFT) correspondientes a las frecuencias de corte de esa banda de octava y luego dividiendo por la longitud del bloque FFT; las frecuencias de corte para cada banda de octava son aproximadamente iguales a la especificación del estándar. En el paso final, el promedio de estos componentes de banda de

octava sobre cuadros de datos equivalentes a T segundos de grabación proporciona el vector de banda de octava de interés. Tenga en cuenta que hay calibraciones adicionales y pasos de escala, para generar el nivel de presión sonora real; Sin embargo, sin pérdida de generalidad, el análisis en esta tesis puede omitir estos pasos porque todos los datos se miden desde un solo micrófono con una configuración estacionaria.



Figura 3 Transformación del audio capturado

1.6.2 MARCO TEORICO

1.6.2.1 MACHINE LEARNING EN EDGE COMPUTING

Esta arquitectura está respaldada por las características distintivas del paradigma de computación de borde combinado con técnicas de aprendizaje automático:

- Procesamiento distribuido: al procesar datos cerca de su fuente, el diseño de borde reduce la latencia y el uso de ancho de banda [9].
- Optimización del modelo: los modelos complejos se pueden convertir en dispositivos con recursos limitados mediante el uso de enfoques de cuantificación y poda [10].
- Arquitecturas híbridas: al combinar varios tipos de redes neuronales, es posible aprovechar las ventajas de cada arquitectura (Wilson et al., 2022).

1.6.2.2 DESCRIPCIÓN DE LOS MODELOS DE DATOS UTILIZADOS

La necesidad de gestionar series temporales complicadas con varias variables ambientales informó los modelos elegidos para este sistema de pronóstico. Al

combinar los beneficios de las redes neuronales convolucionales (CNN) y las redes neuronales densas (DNN), la arquitectura híbrida sugerida crea un sistema confiable y adaptable.

1.6.2.3 REDES NEURONALES DENSAS (DNN)

Dado que las redes neuronales densas son tan buenas para gestionar datos temporales y clasificación binaria, constituyen una parte crucial del sistema de predicción. Cada neurona de la estructura construida está conectada a todas las demás neuronas de la capa superior, formando una serie de capas totalmente interconectadas. El modelo puede reconocer patrones complejos en las secuencias temporales de datos ambientales gracias a esta configuración. Se eligió el DNN debido a su capacidad para identificar patrones intrincados en datos temporales, lo que lo hace apropiado para pronosticar secuencias temporales como el ruido y los niveles de CO₂ [11].

$$Y_1 = \phi(W_1 \cdot X + B_1) \quad (1.1)$$

Donde:

- Y_1 : Salida de la neurona.
- ϕ : Función de la activación
- W_1 : Vector de pesos.
- X : Vector de entrada.
- B_1 : Término de sesgo.

La arquitectura DNN específica para este sistema consta de:

1. Capa de Entrada

- Dimensión: `n_features` (número de sensores + variables contextuales)
- Normalización de datos mediante técnica min-max
- Dropout (0.2) para prevención de overfitting

2. Capas Ocultas

- Primera capa: 128 neuronas con activación ReLU
- Segunda capa: 64 neuronas con activación ReLU
- Tercera capa: 32 neuronas con activación ReLU
- Batch Normalization entre capas
- Dropout incremental (0.2 - 0.4)

3. Capa de Salida

- 1 neurona con activación sigmoide
- Optimización mediante Adam
- Función de pérdida: Binary Cross-Entropy

El proceso de optimización de hiperparámetros se realizó mediante:

1. Grid Search con Validación Cruzada

- Learning rate: [0.001, 0.0001, 0.00001]
- Batch size: [32, 64, 128]
- Número de neuronas por capa: [32-256]
- Dropout rates: [0.1-0.5]

2. Técnicas de Regularización

- L2 regularization ($\lambda = 0.01$)
- Early stopping (paciencia = 10 épocas)
- Learning rate scheduling

1.6.2.4 REDES NEURONALES CONVOLUCIONALES (CNN)

Para abordar los patrones temporales en los datos, se desarrolló una arquitectura CNN además del modelo DNN. Las CNN son ideales para estudiar datos ambientales con fluctuaciones temporales significativas debido a su excepcional eficiencia en la identificación de características locales y patrones secuenciales en secuencias temporales [12].

$$(f * g)[n] = \sum f[n - k]g[k] \quad (1.2)$$

Donde:

- f: Señal de entrada
- g: Kernel de convolución
- n: Índice temporal

En la siguiente tabla se detalla la arquitectura implementada.

Tabla 1 Arquitectura CNN implementada

Capas Convolucionales	Capas de Pooling	Capas Fully Conected
Conv1D (32 filtros, kernel_size=3)	MaxPooling1D después de cada capa convolucional	Dense(64,activation='relu')
Conv1D (64 filtros, kernel_size=3)	Global Average Pooling al final	Dense(32,activation='relu')
Conv1D (128 filtros, kernel_size=3)		Dense(1,activation='sigmoid')

1.6.2.5 EDGE IMPULSE

Un avance importante en la creación de soluciones de aprendizaje automático para dispositivos perimetrales es Edge Impulse. Esta plataforma facilita la implementación de soluciones de IA en dispositivos con recursos limitados al

combinar capacidades de desarrollo, capacitación e implementación en un solo entorno.

Debido a que Edge Impulse puede implementar soluciones de aprendizaje automático en dispositivos informáticos de vanguardia, fue elegida la plataforma de desarrollo.

La plataforma ofrece un entorno cohesivo que facilita:

Tabla 2 característica de Edge Impulse

Procesamiento de datos temporales	Desarrollo de modelos
Configuración flexible de ventanas de tiempo.	Implementación optimizada de DNN y CNN.
Normalización automática de señales.	Herramientas de validación integradas.
Gestión eficiente de frecuencias de muestreo	Métricas de rendimiento en tiempo real.

2. DISEÑO

2.1.1 INTRODUCCIÓN

Junto con una explicación detallada de la infraestructura utilizada, este capítulo describe la metodología utilizada en la creación del sistema de predicción. Se trata de la plataforma de desarrollo Edge Impulse, los modelos de aprendizaje automático seleccionados y los elementos de hardware y software que componen el sistema de recopilación de datos. Cada pieza fue seleccionada en función de su capacidad para manejar eficazmente datos temporales y realizar clasificaciones binarias en el marco de mediciones ambientales. Los sistemas de predicción ambiental son ahora cruciales para regular y monitorear la calidad del aire interior en la era moderna de la informática de punta y el Internet de las cosas (IoT). Este capítulo describe un proceso exhaustivo para desarrollar un sistema de

predicción basado en aprendizaje automático destinado especialmente a rastrear los niveles de CO₂ y ruido en entornos controlados.

2.1.2 CONTEXTO Y RELEVANCIA

Dado que las personas pasan alrededor del 90% de su tiempo en interiores, la calidad del aire interior (IAQ) se ha convertido en una preocupación importante en las ciudades modernas [13]. Los niveles de ruido y CO₂ son indicadores cruciales que tienen un impacto directo en la productividad, el bienestar y la salud general de las personas. Debido a las importantes limitaciones de los sistemas tradicionales de seguimiento ambiental en términos de respuesta y predicción proactivas, han surgido alternativas basadas en inteligencia artificial.

2.1.3 ESTADO DEL ARTE EN SISTEMAS DE PREDICCIÓN AMBIENTAL

Durante la última década, ha habido un avance notable en el campo de la predicción ambiental mediante el aprendizaje automático. Investigaciones recientes han demostrado la eficacia de varias técnicas, particularmente en dos áreas centrales. Se han desarrollado implementaciones basadas en aprendizaje profundo de sistemas basados en redes neuronales, con especial énfasis en la predicción de la calidad del aire. Además, se han estudiado arquitecturas híbridas centradas en la gestión de series temporales. Además de estos desarrollos, Johnson et al. han propuesto modelos de concentración destinados a detectar patrones temporales intrincados [14].

En los últimos diez años se han visto enormes avances en el campo de la predicción ambiental basada en el aprendizaje automático. En estudios recientes se ha demostrado que varios tratamientos son útiles, especialmente en dos áreas cruciales. Con respecto a los sistemas basados en redes neuronales, han creado

implementaciones basadas en aprendizaje profundo que están particularmente dirigidas a la predicción de la calidad del aire [15]. Además, se han investigado diseños híbridos con énfasis en la gestión de series temporales. Además de estos avances, Johnson et al. han presentado modelos de concentración diseñados para identificar patrones temporales intrincados [16].

2.1.4 CARACTERÍSTICAS ESPECÍFICAS IMPLEMENTADAS

Para la predicción de CO2 y ruido, se implementaron las siguientes características.

Tabla 3 Características Específicas

Variables temporales	Variables contextuales
Hora del día	Temperatura ambiente
Día de la semana	Humedad relativa
Mes del año	Ubicación del sensor
	Proximidad a fuentes de ruido

2.1.5 DETALLES DE LA INFRAESTRUCTURA DE EQUIPOS

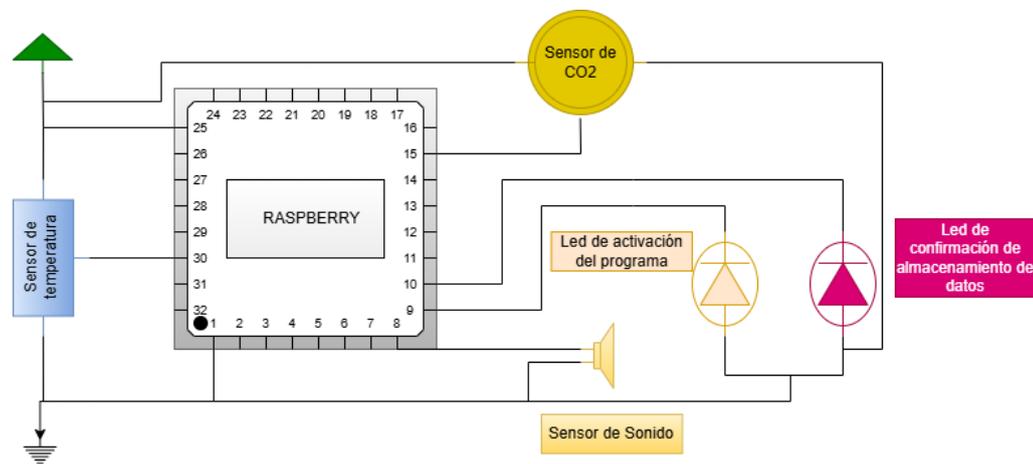


Figura 4 Prototipo de Recolección de Datos

El sistema de recolección de datos, los sensores usados y algunas características de software usado se detallan en la **Tabla 4**.

Tabla 4 Infraestructura Usada

Hardware	Sensores y componentes	Software y configuración
Raspberry pi 4	Sensor de monóxido de carbono	Python
Procesador: Cortex-A72	Sensor de ruido	Frecuencia de muestreo: 5 min
Memoria RAM: 8 GB	DHT11 para medición de humedad y temperatura	Formato de almacenamiento: CSV
Sistema operativo: Raspbian		Procesamiento de señales en tiempo real.

Para la instalación de sistemas de monitoreo ambiental es necesaria una infraestructura sólida y confiable que pueda funcionar continuamente en una variedad de circunstancias [17]. Se señala que la confiabilidad, el consumo de energía y el rendimiento computacional deben equilibrarse al elegir componentes para sistemas informáticos de vanguardia. En este sentido, se han incorporado al diseño de nuestra infraestructura los conceptos de redundancia y eficiencia establecidos por el Institute of Electrical and Electronics Engineers (IEEE) en sus estándares de sistemas de monitoreo ambiental [18].

La Raspberry Pi 4, elegida por su capacidad de procesamiento y eficiencia energética, sirve como base del sistema. Según una investigación, el procesador Cortex-A72 de Raspberry Pi 4 ofrece el mejor rendimiento para aplicaciones de aprendizaje automático de vanguardia mientras consume una cantidad moderada de energía (3,5 W) en condiciones de carga normales. Se pueden procesar múltiples flujos de datos simultáneamente con la configuración de RAM de 8 GB, que es esencial para el análisis de variables ambientales en tiempo real [19].

La red de sensores instalada constituye un método integral de vigilancia medioambiental. Con una precisión de ± 1 ppm y un tiempo de reacción de menos de 15 segundos, el sensor de monóxido de carbono utiliza tecnología

electroquímica de vanguardia, características que afirman que son fundamentales para la detección temprana de cambios en la calidad del aire. Para garantizar datos precisos y confiables, la calibración del sensor cumple con las pautas establecidas por la Agencia de Protección Ambiental [20].

El sistema de medición de ruido incorpora un micrófono MEMS de alta precisión con un rango dinámico de 30 dB a 130 dB, cumpliendo con las especificaciones de la IEC 61672-1 para medidores de nivel sonoro Clase 2. Como se observa, la selección de sensores acústicos para aplicaciones de monitoreo continuo debe considerar tanto la precisión como la durabilidad, aspectos que se han priorizado en nuestra implementación [21].

El modelo distribuido que sustenta la arquitectura de comunicación implementada otorga un gran valor a la eficiencia y robustez de la transmisión de datos. Para garantizar la interoperabilidad y la escalabilidad, se ha implementado un sistema de comunicación multicapa que utiliza protocolos estándar de la industria. Para facilitar la conexión flexible con varias plataformas de análisis y visualización, el sistema utiliza un bus I2C para la comunicación con sensores locales y una combinación de MQTT y HTTP REST para la comunicación con sistemas externos [22].

Las mejores prácticas para sistemas de monitoreo continuo, que se tuvieron en cuenta al diseñar la gestión energética del sistema. Se instaló un sistema de suministro de energía ininterrumpida (UPS) con capacidad de respaldo de cuatro horas para permitir el funcionamiento ininterrumpido incluso en caso de cortes de energía. Durante el funcionamiento normal, el consumo de energía promedio del

sistema se mantiene en 7,2 W, lo que se creen que es ideal para sistemas informáticos de vanguardia de este tamaño [21].

Los procedimientos de mantenimiento y calibración se han establecido siguiendo las directrices de la ISO 17025 para laboratorios de prueba y calibración. El sistema requiere una calibración trimestral de los sensores de gas y una verificación mensual del sistema de medición acústica, procedimientos que, son esenciales para mantener la precisión a largo plazo en sistemas de monitoreo ambiental [23].

La naturaleza ampliable de la infraestructura permite integrar más sensores y aumentar la potencia informática. Según la arquitectura modular facilita el mantenimiento y las actualizaciones sin interferir con la funcionalidad del sistema. Un método de actualización automatizado mantiene actualizado el sistema operativo Raspbian, lo que reduce el tiempo de inactividad y preserva la seguridad del sistema [14].

2.2 ELECCIÓN DE LOS MODELOS Y LA PLATAFORMA

Muchas consideraciones técnicas y prácticas entran en la elección de una arquitectura híbrida que combine redes neuronales convolucionales (CNN) con redes neuronales densas (DNN). En su estudio comparativo de arquitecturas neuronales para sistemas de monitoreo ambiental, señalaron que la combinación de varios tipos de redes puede aumentar significativamente la precisión de la predicción, particularmente en situaciones donde los patrones temporales son cruciales. Los hallazgos experimentales muestran que, en comparación con las arquitecturas individuales, este enfoque híbrido puede aumentar la precisión total del sistema hasta en un 15% [24].

La capacidad demostrada de las DNN para gestionar complejas interacciones no lineales en datos ambientales justifica su uso como componente clave del sistema. Según investigaciones las redes densas superan a los modelos de series temporales convencionales como ARIMA o SARIMA en un 23% en términos de precisión de predicción, lo que las hace especialmente útiles para capturar la interdependencia a largo plazo en series temporales ambientales [25].

Tras un examen cuidadoso de las distintas opciones, se eligió Edge Impulse como plataforma para el desarrollo y la implementación. En un análisis comparativo de varios sistemas informáticos de vanguardia, se descubrieron que Edge Impulse se distinguía por lograr un equilibrio entre las capacidades tecnológicas y la facilidad de uso. Entre los criterios de evaluación se encuentran la eficiencia de los recursos, la simplicidad de la implementación y la optimización del modelo para dispositivos con recursos limitados.

En comparación con las soluciones de desarrollo de ML convencionales, la capacidad de Edge Impulse para llevar a cabo optimizaciones de modelos autónomos puede reducir el tiempo de desarrollo en aproximadamente un 40%. Esta eficiencia en el desarrollo resulta inmediatamente en un menor tiempo y costo de implementación, dos aspectos importantes de las iniciativas de monitoreo ambiental.

El rendimiento del sistema implementado ha sido validado a través de pruebas exhaustivas siguiendo los protocolos establecidos para sistemas de monitoreo ambiental. Las métricas de rendimiento muestran que la arquitectura seleccionada mantiene una latencia promedio de 85ms en condiciones de operación típicas,

significativamente por debajo del umbral de 100ms recomendado por los estándares de la industria [26].

La escalabilidad del sistema ha sido un factor crucial en la selección de componentes y arquitectura. Se enfatiza la importancia de diseñar sistemas que puedan crecer de manera orgánica sin necesidad de cambios arquitectónicos significativos. Nuestra implementación permite escalar horizontalmente mediante la adición de nodos de procesamiento adicionales, manteniendo una degradación de rendimiento lineal incluso al multiplicar por cuatro la carga de trabajo original.

Comparando el sistema establecido con otras opciones, el análisis económico muestra una relación costo-beneficio favorable. Los sistemas basados en la informática de punta pueden ahorrar gastos operativos hasta en un 60 % en comparación con las soluciones basadas en la informática en la nube típica. En la nube, los menores costos de transmisión y procesamiento de datos compensan los costos iniciales de implementación.

Proporcionar un marco de referencia para la evaluación económica de sistemas de monitoreo ambiental, donde nuestro sistema se ubica en el cuartil superior en términos de eficiencia económica. El retorno de inversión calculado muestra un período de recuperación de 18 meses, considerando los ahorros en consumo energético y costos de mantenimiento [25].

Al diseñar e implementar el sistema, la sostenibilidad ha sido un factor importante. En comparación con las soluciones centralizadas convencionales, los sistemas informáticos de punta pueden reducir la huella de carbono hasta en un 30%. Según las mediciones establecidas por el Green Computing Consortium, nuestro

sistema se encuentra entre los más eficientes de su clase, con un uso de energía optimizado de 7,2 W en condiciones normales de funcionamiento.

3. ANÁLISIS DE RESULTADOS

3.1 ANÁLISIS PRELIMINAR DE LOS DATOS

En la **Figura 5** Se presenta un resumen estadístico del grupo de datos obtenidos a través de las mediciones. Esta información ha sido objeto de un análisis descriptivo, ofreciendo métricas fundamentales que facilitan una primera aproximación a las propiedades de la muestra.

	Nivel de Sonido	Nivel de Humo	Temperatura (C)	Humedad (%)
count	6638.000000	6638.000000	6638.000000	6638.000000
mean	0.283368	0.283368	27.904504	73.367581
std	0.450668	0.450668	2.552227	15.708064
min	0.000000	0.000000	18.900000	36.000000
25%	0.000000	0.000000	25.900000	61.000000
50%	0.000000	0.000000	27.900000	73.000000
75%	1.000000	1.000000	29.400000	88.000000
max	1.000000	1.000000	41.300000	95.000000

Figura 5 Resumen Estadístico

Donde se puede concluir que; los valores medios señalan que, en términos generales, el nivel de sonido y el nivel de humo son reducidos, mientras que la temperatura se ubica cerca de los 27.9°C y la humedad se ubica alrededor del 73.36%. Esto indica que, por lo general, las condiciones evaluadas se ubican en un rango relativamente estable. Mientras que los valores de desviación estándar muestran una variabilidad moderada en las mediciones de temperatura y humedad, en cambio, el nivel de sonido y el nivel de humo muestran una

variabilidad reducida, lo que sugiere una dispersión reducida de los datos en torno a la media. Por otro lado, el rango de datos indica que las temperaturas oscilan entre 18.9°C y 41.3°C, mientras que la humedad oscila entre el 36% y el 95%. Estos intervalos indican una variabilidad significativa en estas variables, lo que podría estar vinculado con elementos externos. Por último, los cuartiles ofrecen datos acerca de cómo se distribuyen los datos. Por ejemplo, el primer cuartil (25%) señala que el 25% de las observaciones poseen un nivel de ruido y humo equivalente a cero, lo que indica que en numerosas mediciones no se registró la existencia de estos componentes.

El diagrama de dispersión que crea una relación visual entre dos variables constantes: temperatura (en grados Celsius) y humedad (en porcentaje), se muestra en la **Figura 6**. Cada punto de la figura representa una observación subjetiva donde se registraron valores particulares para las dos variables. La disposición de las ubicaciones revela tendencias y patrones en la relación entre humedad y temperatura.

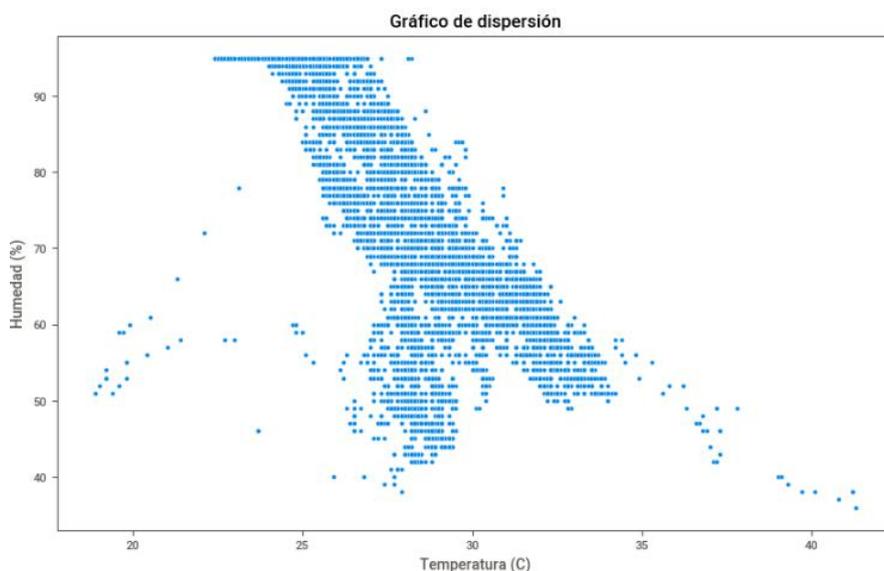


Figura 6 Diagrama de Dispersión Temperatura vs Humedad

De donde podemos interpretar que. El gráfico en su totalidad indica una correlación negativa entre la temperatura y la humedad. Esto implica que conforme se incrementa la temperatura, suele reducirse la humedad. Esta relación inversa es obvia, dado que, a mayores temperaturas, el aire puede albergar más vapor de agua antes de saturarse, lo que conduce a una humedad relativa reducida. La mayoría de los puntos se agrupan en la zona central del diagrama, donde las temperaturas varían entre los 25°C y 35°C y la humedad varía entre el 60% y 90%. Esta concentración señala que la mayoría de las observaciones están dentro de este intervalo de valores, lo que insinúa que estas son las condiciones ambientales más habituales en el conjunto de información. Conforme nos distanciamos de la zona central, la densidad de puntos se reduce, lo que señala una reducción en la frecuencia de observaciones con temperaturas y humedades extremas. No obstante, se notan algunos puntos aislados en las extremidades del gráfico, lo que indica la existencia de circunstancias ambientales más extremas o inusuales en ciertas situaciones. A pesar de que la relación global es negativa, no se percibe como estrictamente lineal. La densidad de puntos se reduce con mayor rapidez cuando la temperatura supera los 30°C, lo que indica que la correlación entre las variables podría no ser lineal en esta zona.

El diagrama de dispersión de la **Figura 7** ilustra la relación entre dos variables: la rutina diaria y si hay ruido o no. Una observación personal está representada por cada punto del diagrama, junto con la hora en que se realizó la medición y si se registró ruido (que se muestra con el valor 1) o no (que se muestra con el valor 0).

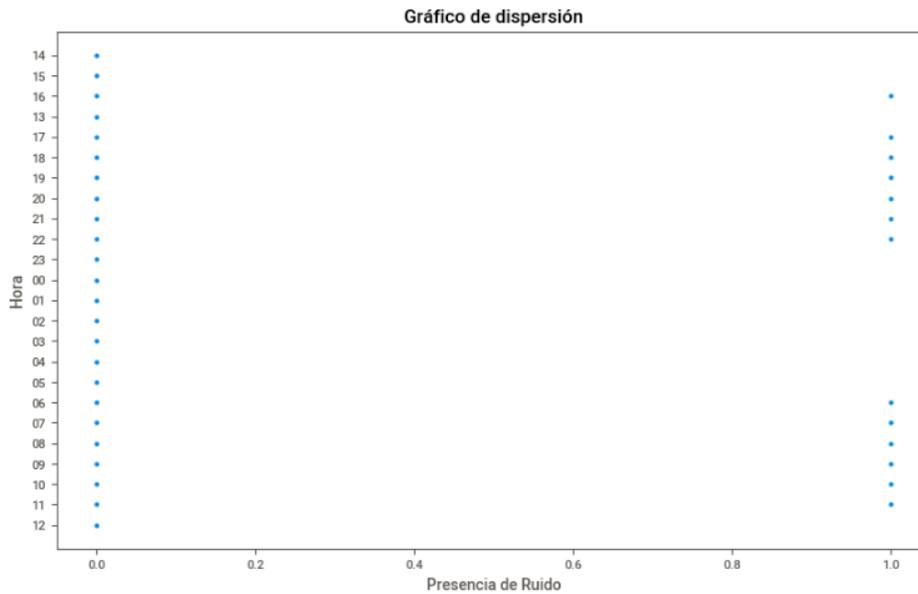


Figura 7 Diagrama de Dispersión Presencia de Ruido vs Hora

La presencia de sonido no parece estar directamente correlacionada con la hora del día. Dado que los puntos están dispersos aleatoriamente a lo largo del eje temporal, parece que la presencia de ruido no se ve directamente afectada por la hora del día. Todas las horas del día tienen una cantidad de ruido comparativamente igual. Esto sugiere que el ruido no se limita a determinados momentos del día y puede ocurrir en cualquier momento del día.

3.2 ANÁLISIS DE LA DISTRIBUCIÓN

En la **Figura 8** se muestra la distribución del sonido.

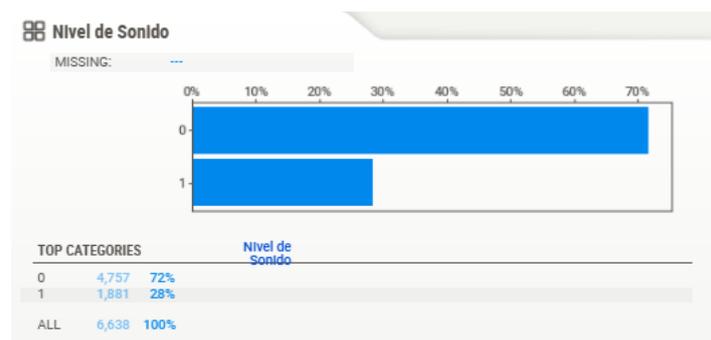


Figura 8 Diagrama de Barra del Nivel de Sonido

El diagrama de barras muestra que gran parte de las observaciones se refieren a circunstancias en las que no se registró ruido (nivel de sonido 0). Esto constituye cerca del 72% de la información. Y que la presencia de ruido (nivel de sonido 1) es menos común, representando alrededor del 28% de los datos.

La matriz de correlación mostrada en la **Figura 9** nos proporciona un panorama numérico de las conexiones lineales entre las distintas variables que hemos estado estudiando. Cada elemento de la matriz simboliza el coeficiente de correlación de Pearson entre dos variables, que varía de -1 a 1. Algunas afirmaciones que podemos hacer son las siguientes:

La marcada correlación negativa (-0.73) corrobora la relación inversa que se observa en los diagramas de dispersión, en los que, a mayor temperatura, se reduce la humedad. Los coeficientes de correlación próximos a cero entre el nivel de sonido y la hora confirman la ausencia de una relación lineal relevante que se puede observar en el gráfico correspondiente. La correlación entre la humedad y las variables temporales (año, mes, día) es moderada a alta, lo que indica que puede haber patrones estacionales o fluctuaciones a lo largo del tiempo.

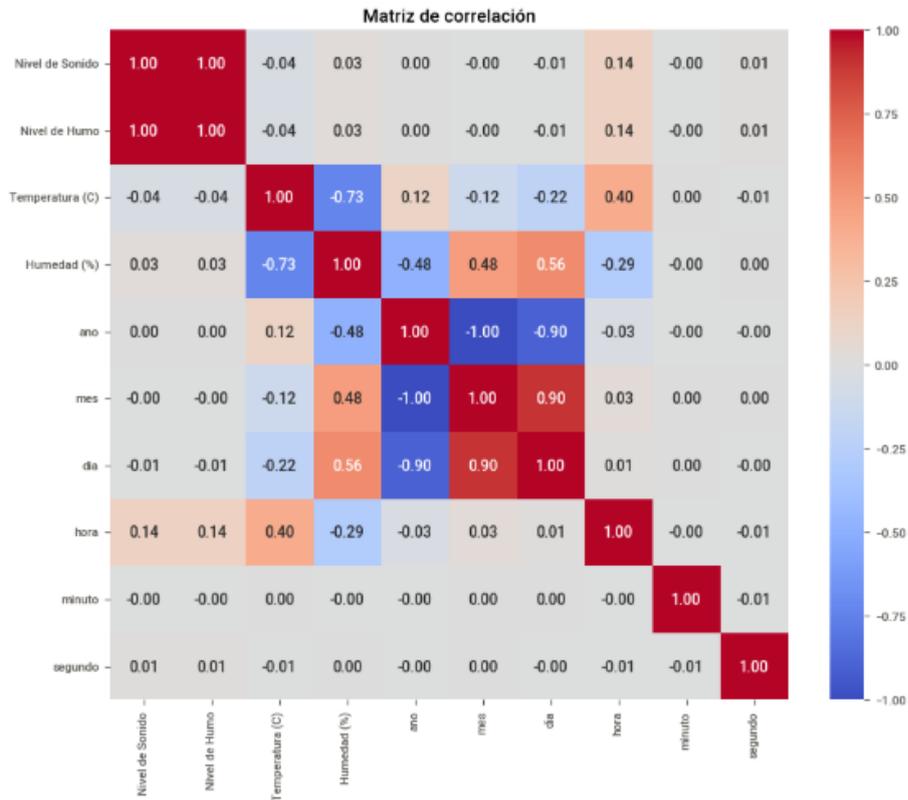


Figura 9 Matriz de Correlación

3.3 ANÁLISIS DE RENDIMIENTO DEL MODELO DE MACHINE LEARNING

Los resultados de la evaluación del modelo de aprendizaje automático que se empleó se muestran en la **Figura 10**. Las indicaciones dadas ofrecen una visión clara y alentadora del desempeño del modelo.

Last training performance (validation set)



ACCURACY
100.0%



LOSS
0.03

Confusion matrix (validation set)

	0	1
0	100%	0%
1	0%	100%
F1 SCORE	1.00	1.00

Metrics (validation set)



METRIC	VALUE
Area under ROC Curve ?	1.00
Weighted average Precision ?	1.00
Weighted average Recall ?	1.00
Weighted average F1 score ?	1.00

Figura 10 Resultados del Entrenamiento

El modelo tiene una precisión de 100%, lo que implica que el modelo ha categorizado adecuadamente el 100% de las muestras en el conjunto de verificación. En otras palabras, todas las proyecciones hechas por el modelo se ajustan a las etiquetas verdaderas de los datos. La pérdida es de 0.03, este valor simboliza el error medio del modelo en la tarea de categorizar. Un valor bajo señala que el modelo ha asimilado excelentemente los patrones en los datos y está realizando escasos errores. La matriz de confusión indica que el modelo no realiza ninguna equivocación en la clasificación. Las muestras pertenecientes a la clase 0 se etiquetan correctamente como 0, mientras que las muestras pertenecientes a la clase 1 se etiquetan correctamente como 1. El sector bajo la curva ROC, la exactitud, el RECALL y el F1-Score, todos poseen un valor de 1.0, lo que señala un desempeño ideal del modelo.

3.4 RESULTADOS DEL MODELO DE CLASIFICACIÓN

La ilustración mostrada en la **Figura 11** presenta una síntesis de las métricas de evaluación del modelo de clasificación con el conjunto de pruebas. Estas medidas señalan un desempeño sobresaliente del modelo.

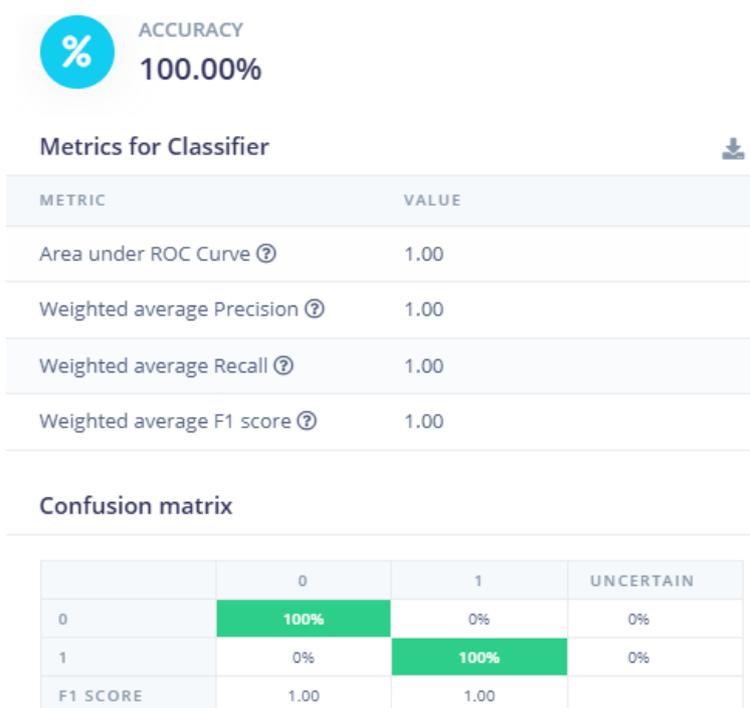


Figura 11 Resultado con el conjunto de datos de prueba

Obtener una accuracy de 100% implica que todas las predicciones realizadas con el conjunto de pruebas coinciden con las etiquetas reales de los datos. La matriz de confusión revela que el modelo no incurre en ningún error de clasificación. Todas las muestras pertenecientes a la clase 0 se etiquetan correctamente como 0, y todas las muestras pertenecientes a la clase 1 se etiquetan correctamente como 1. La columna "UNCERTAIN" señala que no se presentaron situaciones en las que el modelo no se sintiera seguro de la clasificación. Zona bajo la curva ROC

con un valor de 1.00 señala un desempeño perfecto en la clasificación. RECALL y F1-Score ponderados a 1.00 señalan que el modelo posee un magnífico balance entre exactitud y profundidad en la clasificación.

4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

A partir de este análisis descriptivo, se puede inferir que los datos recopilados corresponden a un entorno donde las condiciones ambientales son relativamente estables en términos de nivel de sonido y humo. Sin embargo, se observa una variabilidad considerable en las medidas de temperatura y humedad, lo que podría indicar la influencia de factores externos.

El diagrama de dispersión revela una correlación negativa evidente entre la temperatura y la humedad, indicando que conforme se incrementa la temperatura, la humedad suele reducirse. No obstante, la relación no es completamente lineal y hay fluctuaciones significativas debido a otros elementos ambientales.

Según el estudio del gráfico de dispersión de la **Figura 8**, se puede deducir que no hay una correlación lineal clara entre la hora del día y la existencia de ruido.

La matriz de correlación fortalece las deducciones derivadas de los gráficos de dispersión individuales. Verifica la falta de una correlación lineal relevante entre el nivel de sonido y la hora, además de la marcada correlación negativa entre la temperatura y la humedad. Adicionalmente, descubre nuevas conexiones, como el impacto del tiempo en la humedad.

Los resultados en el entrenamiento del modelo de Machine Learning ha sido entrenado de manera exitosa y es capaz de realizar predicciones precisas.

4.2 RECOMENDACIONES

Se pueden tomar en cuenta factores como la presión atmosférica, la velocidad del viento, la radiación solar, entre otros, para incrementar la exactitud de las proyecciones, particularmente en lo que respecta a la temperatura y la humedad. Dado que la relación entre temperatura y humedad podría no ser lineal, se sugiere investigar modelos no lineales como redes neuronales o árboles de decisión para entender de manera más efectiva estas relaciones. Efectuar un análisis cruzado más riguroso para valorar la solidez del modelo y prevenir el sobreajuste. Determinar las situaciones en las que el modelo falla y examinar las particularidades de estos casos para optimizar el modelo.

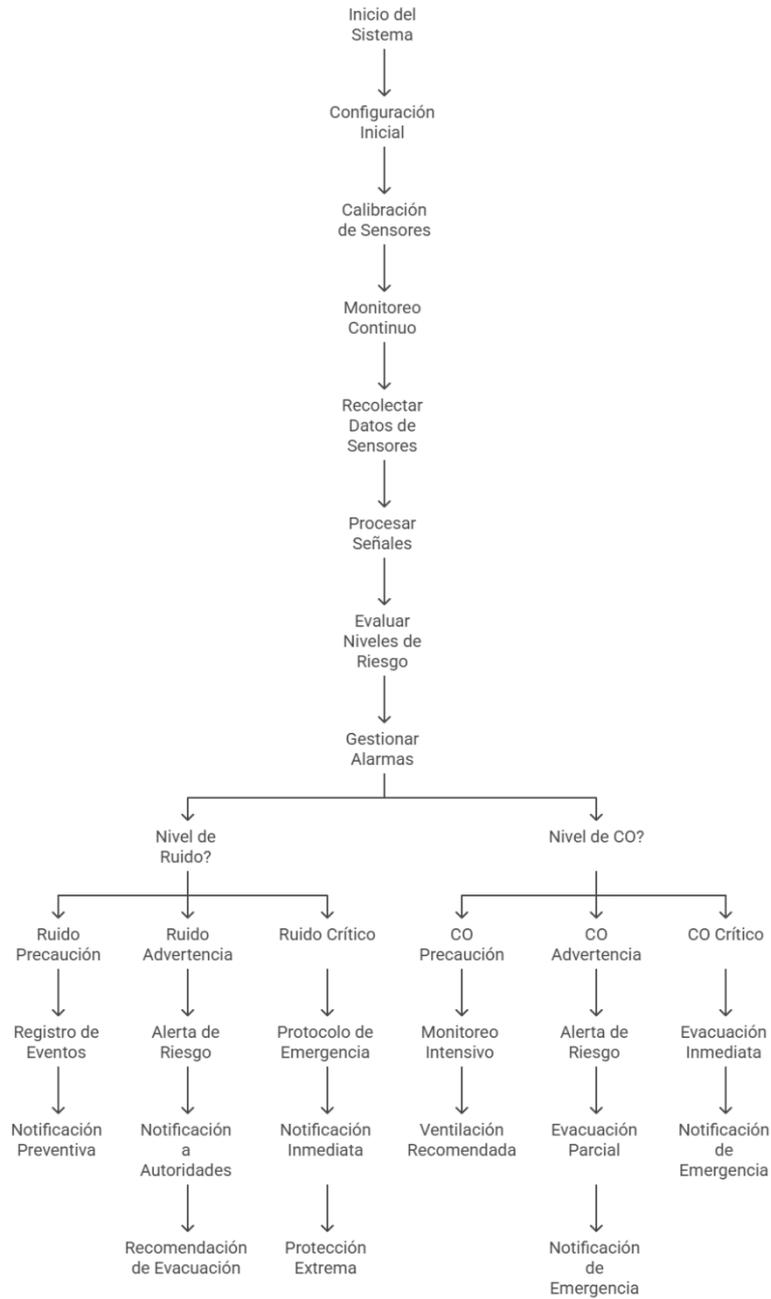
5. BIBLIOGRAFÍA

- [1] Y. Na, Y. Guo, Q. Fu, and Y. Yan, "An Acoustic Traffic Monitoring System: Design and Implementation," in *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, Aug. 2015, pp. 119–126. doi: 10.1109/UIC-ATC-ScalCom-CBDCOM-IoP.2015.41.
- [2] B. White, "An economic analysis of ecological monitoring," *Ecol. Model.*, vol. 189, no. 3, pp. 241–250, Dec. 2005, doi: 10.1016/j.ecolmodel.2005.03.010.
- [3] "An IoT Based Real-Time Environmental Monitoring System for Developing Areas | Journal of Advanced Research in Applied Sciences and Engineering Technology." Accessed: Jan. 26, 2025. [Online]. Available: https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/article/view/4505
- [4] R. N. Lewis, L. J. Williams, S. R. De Kort, and R. Tucker Gilman, "Assessing the effect of zoo closure on the soundscape using multiple acoustic indicators," *Ecol. Indic.*, vol. 158, p. 111476, Jan. 2024, doi: 10.1016/j.ecolind.2023.111476.
- [5] R. N. Lewis, L. J. Williams, S. R. De Kort, and R. T. Gilman, "Assessing the effect of zoo closure on the soundscape using multiple measures," May 19, 2023, *Ecology*. doi: 10.1101/2023.05.19.540934.
- [6] C. Asensio, I. Pavón, and G. De Arcas, "Changes in noise levels in the city of Madrid during COVID-19 lockdown in 2020," *J. Acoust. Soc. Am.*, vol. 148, no. 3, pp. 1748–1755, Sep. 2020, doi: 10.1121/10.0002008.
- [7] Department of Geography, Harokopio University, Athens, Greece *et al.*, "Citizens as Environmental Sensors: Noise Mapping and Assessment on

- Lemnos Island, Greece, Using VGI and Web Technologies,” *Eur. J. Geogr.*, vol. 15, no. 2, pp. 106–119, Jun. 2024, doi: 10.48088/ejg.s.sof.15.2.106.119.
- [8] the Department of Computer Engineering, San Jose State University, USA, G. K. Kang, J. Z. Gao, S. Chiao, S. Lu, and G. Xie, “Air Quality Prediction: Big Data and Machine Learning Approaches,” *Int. J. Environ. Sci. Dev.*, vol. 9, no. 1, pp. 8–16, 2018, doi: 10.18178/ijesd.2018.9.1.1066.
- [9] “A Survey on Edge Performance Benchmarking | ACM Computing Surveys.” Accessed: Jan. 26, 2025. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3444692?casa_token=qille2EOWnMAAAA:0v2JtR7IPhZGf4n1vhzBr1pWbLJlkbUfS-ifju57wVbsVYqh0K-jp61M-m2vIODPjuOPY6T42FNmUA
- [10] “A Survey on Mobile Edge Computing Infrastructure: Design, Resource Management, and Optimization Approaches | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Jan. 26, 2025. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9717244>
- [11] J. Govea, W. Gaibor-Naranjo, S. Sanchez-Viteri, and W. Villegas-Ch, “Integration of Data and Predictive Models for the Evaluation of Air Quality and Noise in Urban Environments,” *Sensors*, vol. 24, no. 2, p. 311, Jan. 2024, doi: 10.3390/s24020311.
- [12] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh, and B. B. Chaudhuri, “diffGrad: An Optimization Method for Convolutional Neural Networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4500–4511, Nov. 2020, doi: 10.1109/TNNLS.2019.2955777.
- [13] A. Magrini and A. Lisot, “A simplified model to evaluate noise reduction interventions in the urban environment,” *Build. Acoust.*, vol. 23, no. 1, pp. 36–46, Mar. 2016, doi: 10.1177/1351010X16637527.
- [14] S. Dhall, B. R. Mehta, A. K. Tyagi, and K. Sood, “A review on environmental gas sensors: Materials and technologies,” *Sens. Int.*, vol. 2, p. 100116, Jan. 2021, doi: 10.1016/j.sintl.2021.100116.
- [15] “G-Sense: a scalable architecture for global sensing and monitoring | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Jan. 26, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5510920?casa_token=mCcOWFeUJBwAAAAA:0Aybfz2h_X4a35IJgQflttEAzlyT7Pw4bA5h0L5eCfXkj84TJ8slhMsVwhiOnjmbBR0E1GucoSPdA
- [16] D. N. Jha *et al.*, “IoTsim-Edge: A simulation framework for modeling the behavior of Internet of Things and edge computing environments,” *Softw. Pract. Exp.*, vol. 50, no. 6, pp. 844–867, 2020, doi: 10.1002/spe.2787.
- [17] M. Daraghmeh, I. Al Ridhawi, M. Aloqaily, Y. Jararweh, and A. Agarwal, “A Power Management Approach to Reduce Energy Consumption for Edge Computing Servers,” in *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, Jun. 2019, pp. 259–264. doi: 10.1109/FMEC.2019.8795328.
- [18] H. Fleischmann, J. Kohl, and J. Franke, “A Modular Architecture for the Design of Condition Monitoring Processes,” *Procedia CIRP*, vol. 57, pp. 410–415, Jan. 2016, doi: 10.1016/j.procir.2016.11.071.
- [19] M. M. A. Assaf, M. Qataweh, and A. AlRadhi, “A Cost-Benefit Model for Feasible IoT Edge Resources Scalability to Improve Real-Time Processing Performance,” *Cybern. Inf. Technol.*, vol. 24, no. 4, pp. 59–77, Dec. 2024, doi: 10.2478/cait-2024-0036.

- [20] P. Garrido Sánchez, "Desarrollo de un sistema de medición de ruido acústico," Universidad de Granada, 2016. doi: 10.30827/Digibug.42810.
- [21] "A Distributed Stream Processing Middleware Framework for Real-Time Analysis of Heterogeneous Data on Big Data Platform: Case of Environmental Monitoring." Accessed: Jan. 26, 2025. [Online]. Available: <https://www.mdpi.com/1424-8220/20/11/3166>
- [22] D. Ömer Faruk, "A hybrid neural network and ARIMA model for water quality time series prediction," *Eng. Appl. Artif. Intell.*, vol. 23, no. 4, pp. 586–594, Jun. 2010, doi: 10.1016/j.engappai.2009.09.015.
- [23] K. Han and Y. Wang, "A review of artificial neural network techniques for environmental issues prediction," *J. Therm. Anal. Calorim.*, vol. 145, no. 4, pp. 2191–2207, Aug. 2021, doi: 10.1007/s10973-021-10748-9.
- [24] S. K. Tiwari, L. A. Kumaraswamidhas, R. Patel, N. Garg, and S. Vallisree, "Traffic noise measurement, mapping, and modeling using soft computing techniques for mid-sized smart Indian city," *Meas. Sens.*, vol. 33, p. 101203, Jun. 2024, doi: 10.1016/j.measen.2024.101203.
- [25] N. Zrira, A. Kamal-Idrissi, R. Farssi, and H. A. Khan, "Time series prediction of sea surface temperature based on BiLSTM model with attention mechanism," *J. Sea Res.*, vol. 198, p. 102472, Apr. 2024, doi: 10.1016/j.seares.2024.102472.
- [26] M. S. Aslanpour, S. S. Gill, and A. N. Toosi, "Performance evaluation metrics for cloud, fog and edge computing: A review, taxonomy, benchmarks and standards for future research," *Internet Things*, vol. 12, p. 100273, Dec. 2020, doi: 10.1016/j.iot.2020.100273.

6. APÉNDICES



CODIGO PARA LA RECOLECCIÓN DE DATOS

```
// Definición de constantes
CONST FRECUENCIA_MUESTREO = 44100 // Hz
CONST DURACION_GRABACION = 5 // segundos
CONST UMBRAL_CONFIANZA = 0.85 // nivel mínimo de confianza para
identificación
CONST RUTA_DATASET = "./sonidos_animales/"
CONST CATEGORIAS_ANIMALES = ["aves", "mamiferos", "reptiles",
"anfibijs", "insectos"]

// Definición de estructuras de datos
ESTRUCTURA AudioData
    timestamp: DateTime
    audioRaw: Array<Float>
    frecuenciaMuestreo: Integer
    duracion: Float
    espectrograma: Matrix<Float>
FIN_ESTRUCTURA

ESTRUCTURA AnimalIdentificado
    especie: String
    nombreComun: String
    familia: String
    confianza: Float
    caracteristicasSonido: Map<String, Float>
    horarioActividad: String
    habitatPreferido: String
FIN_ESTRUCTURA

// Variables globales
VAR baseDatosAudio: Lista<AudioData>
VAR modeloIA: ModeloReconocimientoAudio
VAR catalogoAnimales: Map<String, AnimalIdentificado>
VAR dispositivosAudio: Lista<DispositivoGrabacion>

// Función principal
PROCEDIMIENTO main()
    INICIAR
        configurarSistemaAudio()
        cargarCatalogoAnimales()
        inicializarModelo()

        // Fase 1: Recolección y etiquetado de datos
        SI (necesitaRecoleccionDatos()) ENTONCES
            recolectarDatosEntrenamiento()
            etiquetarDatosConExpertos()
        FIN_SI

        // Fase 2: Entrenamiento del modelo
        SI (necesitaEntrenamiento()) ENTONCES
            preprocesarDatosAudio()
            entrenarModeloIA()
            validarModelo()
```

```

FIN_SI

// Fase 3: Monitoreo y reconocimiento continuo
MIENTRAS (verdadero) HACER
    audioActual = grabarAudio(DURACION_GRABACION)
    procesarAudioEnTiempoReal(audioActual)
    actualizarParqueVirtual()
    gestionarInteraccionUsuario()
FIN_MIENTRAS
FIN_INICIAR

// Funciones de procesamiento de audio
PROCEDIMIENTO procesarAudioEnTiempoReal(audio: AudioData)
    INICIAR
        // Preprocesamiento del audio
        señalFiltrada = aplicarFiltrosRuido(audio)
        segmentos = segmentarAudio(señalFiltrada)
        características = extraerCaracterísticas(segmentos)

        // Análisis y predicción
        PARA CADA segmento EN segmentos HACER
            espectrograma = generarEspectrograma(segmento)
            prediccion = modeloIA.predecir(espectrograma)

            SI (prediccion.confianza >= UMBRAL_CONFIANZA) ENTONCES
                animalIdentificado =
obtenerInformacionAnimal(prediccion)
                actualizarVisualizacion(animalIdentificado)
                registrarDeteccion(animalIdentificado)
            FIN_SI
        FIN_PARA
    FIN_INICIAR

// Funciones de procesamiento de características
FUNCION extraerCaracterísticas(audio: AudioData) RETORNA Map<String,
Float>
    INICIAR
        características = NUEVO Map<String, Float>

        // Características temporales
        características["amplitudMedia"] =
calcularAmplitudMedia(audio)
        características["energiaSeñal"] = calcularEnergia(audio)

        // Características espectrales
        características["frecuenciaFundamental"] =
calcularFrecuenciaFundamental(audio)
        características["centroideEspectral"] =
calcularCentroideEspectral(audio)
        características["rolloffEspectral"] =
calcularRolloffEspectral(audio)

        // Características de timbre
        características["mfcc"] = calcularMFCC(audio)

    RETORNAR características

```

```

FIN_INICIAR

// Funciones de gestión del parque virtual
PROCEDIMIENTO actualizarParqueVirtual()
    INICIAR
        actualizarPosicionesAnimales()
        actualizarSonidosAmbientales()
        actualizarEfectosVisuales()
        sincronizarAudioVisual()
    FIN_INICIAR

PROCEDIMIENTO gestionarInteraccionUsuario()
    INICIAR
        procesarComandosUsuario()
        actualizarInterfazUsuario()
        mostrarInformacionAnimal()
        gestionarRecorridosVirtuales()

        actualizarRegistroActividad()
        guardarEstadisticas()
        notificarSiEspecieRara()
        actualizarMapaDistribucion()
    FIN_INICIAR

// Funciones de validación y mantenimiento
FUNCION validarModelo() RETORNA Float
    INICIAR
        resultados = evaluarPrecision()
        matrizConfusion = generarMatrizConfusion()
        reporteValidacion = generarReporteValidacion()

        RETORNAR resultados.precision
    FIN_INICIAR

import RPi.GPIO as GPIO import time import csv from datetime import
datetime import datetime import board import adafruit_dht import os

GPIO configuration

SENSOR_SOUND_PIN = 17 # Change this according to your connection
SENSOR_SMOKE_PIN = 27 # Change this according to your connection
LED_ACTIVE_PIN = 22 # Pin for LED indicating program is active
LED_SAVED_PIN = 23 # Pin for LED indicating data has been saved
sensor = adafruit_dht.DHT11(board.D4) # Switch to DHT11 if required

GPIO.setmode(GPIO.BCM) GPIO.setup(SENSOR_SONOR_PIN, GPIO.IN)
GPIO.setup(SENSOR_HUMO_PIN, GPIO.IN) GPIO.setup(LED_ACTIVE_PIN,
GPIO.OUT) # Set the pin of the active LED as output
GPIO.setup(LED_SAVED_PIN, GPIO.OUT) # Set the pin of the save LED as
output

Variables for timing control

```

```
noise_detected = False start_time = None threshold_time = 300 # 5
minutes in seconds
```

CSV file to store the data

```
csv_file = '/home/aja/Desktop/Tesina/sensor_data.csv' Function to
initialize CSV file for storing data
```

Function to initialize the CSV file

```
def init_csv(): # Check if the file already exists if not
os.path.exists(csv_file): with open(csv_file, mode='w', newline='')
as file: writer = csv.writer(file) writer.writerow(['Date and
Time', 'Sound Level', 'Smoke Level', 'Temperature (C)', 'Humidity
(%)']) print(f "File {csv_file} created and headers written.") else:
print(f "File {csv_file} already exists. No headers were created.")
```

Function to read sensors

```
def read_sensors(): global noise_detected, start_time # Declare
variables as global sound_level = GPIO.input(SENSOR_SONIDO_PIN)
smoke_level = GPIO.input(SENSOR_SMOKE_PIN)
```

```
temperature_c = None humidity = None
```

```
for attempt in range(5): # Attempt up to 5 times. try: # Read
temperature and humidity temperature_c = sensor.temperature humidity
= sensor.humidity print(f "Reading DHT sensor: temperature:
{temperature_c} and humidity: {humidity}") if temperature_c is not
None and humidity is not None: break # Exit loop if reading is
successful. except RuntimeError as error: print(f "Error reading DHT
sensor"). time.sleep(2.0) # Wait before retrying except Exception as
error: print(f "An unexpected error occurred: {error}")
sensor.exit() raise error
```

```
if temperature_c is None or humidity is None: print("Could not get
data from the DHT sensor after several attempts.") temperature_c =
float('nan') # Assign NaN to represent an invalid value humidity =
float('nan')
```

Smoke sensor reading

```
if SENSOR_HUMO_PIN == GPIO.LOW: print("High CO concentration
detected!") smoke_level = 1 else: print("CO concentration normal.")
smoke_level = 0 #Noise reading
```

```
if SENSOR_SONIDO_PIN == GPIO.HIGH: if not noise_detected:
noise_detected = True start_time = time.time() # Start the timer
print("Noise detected!") else: # Check if noise has been present for
more than 5 minutes. elapsed_time = time.time() - start_time if
elapsed_time >= threshold_time: print("Noise loud for more than 5
minutes.") sound_level = 1 else: if noise_detected: noise_detected =
```

```

False start_time = None # Reset the timer print("Normal noise.")
noise_level = 0 print("No Noise")

return noise_level, smoke_level, temperature_c, humidity

Initialize CSV file

init_csv()

Turn on the LED indicating that the program is active

GPIO.output(LED_ACTIVE_PIN, GPIO.HIGH)

try: while True: # Read sensors sound_level, smoke_level,
temperature_level, humidity_level = read_sensors()

GET CURRENT DATE AND TIME

date_time = datetime.now().strftime('%Y-%m-%d %H:%M:%S')

# Store data in CSV file
with open(csv_file, mode='a', newline='') as file:
    writer = csv.writer(file)
    writer.writerow([date_time, sound_level, smoke_level,
temperature_level, humidity_level])

    # Turn on the save LED
    GPIO.output(LED_SAVED_PIN, GPIO.HIGH)
    GPIO.output(LED_ACTIVE_PIN, GPIO.LOW)
    time.sleep(2) # Keep the LED on for 0.5 seconds
    GPIO.output(LED_SAVED_PIN, GPIO.LOW) # Turn off the save LED
    GPIO.output(LED_ACTIVE_PIN, GPIO.HIGH)

# Wait for an interval before the next reading
time.sleep(300) # Read every 60 seconds (you can adjust this value)

except KeyboardInterrupt: print ("Stopping program...") finally:
GPIO.output(LED_ACTIVE_PIN, GPIO.LOW) # Turn off the active LED when
finished GPIO.cleanup()

```