



APLICACIÓN DE ALGORITMOS EN LA RECONSTRUCCIÓN DE ESTRUCTURAS
RETICULARES EN MODELOS BIM

ALEXANDRA MILENA MONCADA HERNÁNDEZ

Proyecto de grado

Asesor

Ph.D., Luis Fernando Botero Botero

UNIVERSIDAD EAFIT
ESCUELA DE CIENCIAS APLICADAS E INGENIERÍA
MAESTRÍA EN INGENIERÍA
MEDELLÍN
2025

Agradecimiento

En primer lugar, quiero expresar mi gratitud a Dios, sin Él en mi vida nada sería posible.

A mi familia Claudia Hernandez, Michelle y Andrea Moncada por apoyarme constantemente, su acompañamiento y comprensión en cada uno de los proyectos que he realizado han sido siempre fundamentales en mi formación profesional y personal. A Andrea Márquez Torres, por su presencia incondicional, dándome palabras de aliento que me motivan a alcanzar cada reto y dar siempre lo mejor de mí.

A IEB S.A, en especial a Jaime Blandón Díaz por su decidido respaldo y apoyo generoso el cual me ha permitido avanzar en mis estudios.

CONTENIDO

pág.

| | |
|--|----|
| 0. INTRODUCCIÓN..... | 8 |
| 1. PLANTEAMIENTO DEL PROBLEMA | 9 |
| 2. JUSTIFICACIÓN..... | 16 |
| 3. OBJETIVOS..... | 18 |
| 3.1. GENERAL | 18 |
| 3.2. ESPECÍFICOS | 18 |
| 4. MARCO TEÓRICO O MARCO CONCEPTUAL | 19 |
| 4.1. FUNDAMENTOS CONCEPTUALES..... | 20 |
| 4.2. BASES TEÓRICAS Y METODOLOGICAS | 21 |
| 4.3. ANTECEDENTES INVESTIGATIVOS | 22 |
| 4.4. MARCO NORMATIVO Y ESTÁNDARES TECNOLÓGICOS | 25 |
| 5. DISEÑO METODOLÓGICO O METODOLOGÍA | 26 |
| 5.1. ENFOQUE CUANTITATIVO | 26 |
| 5.2. PROCEDIMIENTO METODOLÓGICO | 27 |
| 5.3. TIPO Y DISEÑO DE LA INVESTIGACIÓN | 28 |
| 5.4. SOFTWARE DE ANÁLISIS | 28 |
| 5.5. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE INFORMACIÓN | 28 |
| 5.6. CRITERIOS DE BUSQUEDA | 33 |
| 5.7. VARIABLES DE ESTUDIO..... | 34 |
| 6. RESULTADOS..... | 35 |
| 6.1. DESCRIPCIÓN DE LOS ALGORITMOS..... | 35 |
| 6.2. SELECCIÓN DEL ALGORITMO..... | 39 |
| 6.3. ETAPAS DE APLICACIÓN DEL ALGORITMO..... | 41 |
| 6.3.1. CONVERTOR E57 A XYZ..... | 42 |
| 6.3.2. CONVERTIDOR XYZ A MALLA 3D | 46 |
| 6.3.3. DOCUMENTACIÓN DEL SCRIPT | 51 |
| 6.4. RESUMEN DE RESULTADOS | 58 |

| | |
|-------------------------|----|
| 7. CONCLUSIONES..... | 65 |
| 8. RECOMENDACIONES..... | 67 |
| REFERENCIAS..... | 68 |
| ANEXOS | 76 |

LISTA DE TABLAS

pág.

| | |
|--|----|
| Tabla 1. Resumen del software comercial. | 14 |
| Tabla 2. Resumen procesamiento metodológico. | 27 |
| Tabla 3. Ecuación de búsqueda. | 30 |
| Tabla 4. Resumen resultados de búsqueda. | 31 |
| Tabla 5. Variables de estudio. | 34 |
| Tabla 6. Algoritmos evaluados. | 38 |
| Tabla 7. Carpetas creadas por el script. | 52 |

LISTA DE FIGURAS

pág.

| | |
|---|----|
| Fig. 1. Resultado de un levantamiento con nube de puntos. | 10 |
| Fig. 2. Modelo 3D – Revit, equipo electrico descargador. | 29 |
| Fig. 3. Modelo 3D – Recap, equipo electrico descargador. | 29 |
| Fig. 4. Número de artículos encontrados en las bases bibliográficas. | 31 |
| Fig. 5. Base de datos vs año de búsqueda | 32 |
| Fig. 6. Artículos seleccionados para revisión y análisis | 32 |
| Fig. 7. Red de citas bibliografías. | 33 |
| Fig. 8. Aplicativo web. | 42 |
| Fig. 9. Aplicativo web, carga de archivos formato e57. | 46 |
| Fig. 10. Aplicativo web, procesamiento de los archivos a mallas 3D | 51 |
| Fig. 11. Diagrama de flujo del Script. | 57 |
| Fig. 12. Aplicativo web, procesamiento del script | 58 |
| Fig. 13. Modelo 3D – Revit, equipo eléctrico seccionador. | 59 |
| Fig. 14. Modelo 3D – Nube de puntos, equipo eléctrico seccionador. | 59 |
| Fig. 15. Procesamiento del algoritmo al generar las imágenes. | 60 |
| Fig. 16. Captura de imágenes - entrenamiento del algoritmo, equipo seccionador. | 60 |
| Fig. 17. Reconstrucción del equipo eléctrico, seccionador. | 61 |
| Fig. 18. Resultados probabilísticos - confianza 99.75%, seccionador. | 61 |
| Fig. 19. Resultados gráficos - confianza 99.75%, seccionador. | 63 |
| Fig. 20. Modelo 3D – Revit, equipo eléctrico descargador. | 64 |
| Fig. 21. Modelo 3D – Revit, equipos eléctricos. | 64 |

LISTA DE ANEXOS

pág.

| | |
|---|----|
| Anexo A. READMI | 76 |
| Anexo B. Video_paso_a_paso_instalacion_aplicativo | 76 |
| Anexo C. Captured_images | 76 |
| Anexo D. Archivos_nubes_de_puntos | 76 |
| Anexo E. Equipos_revit | 76 |
| Anexo F. Classification_results | 76 |
| Anexo G. Result_resume | 76 |
| Anexo H. Informe_de_clasificacion | 76 |

Resumen

Crear modelos BIM [1], [2], [3] a partir de nubes de puntos 3D [4], [5], [6], [7], [8] es una actividad cada vez más frecuente en proyectos de construcción, presenta importantes avances gracias a nuevas herramientas como la inteligencia artificial [9], [10], [11], el levantamiento de superficies a partir del uso de equipos, que pueden ubicarse bien sea en tierra o realizar sobre vuelos para identificar áreas de interés. La forma en que se toman las nubes de puntos es muy importante, porque hace referencia a la calidad de la información que posteriormente será utilizada. Las subactividades restantes, tales como el procesamiento e interpretación de objetos corresponde al principal reto que se quiere evaluar con este proyecto, para con el uso de los datos lograr identificar el objeto 3D [12], [13], [14], [15] a que hace referencia y en un software de modelado generar dicho objeto. Para el procesamiento de estas nubes de puntos existen métodos como Machine Learning & Clustering el algoritmo DBSCAN [16]; aun así, hay retos en la identificación de objetos a partir de nubes de puntos sobre todo de objetos que no se refieren a muros, losas, columnas y vigas [17], el caso de interés es utilizar un método que permita identificar objetos 3D para subestaciones eléctricas, donde las estructuras reticulares [18] son una constante, además, equipos de baja, media y alta tensión. Lo que se pretende es evitar pérdidas de información y reducir errores en la reconstrucción de estos objetos [19]. Una vez recopilada la nube de puntos y realizado el proceso de limpieza de dicha nube, se procesan los puntos que se encuentran en dos tipos de formatos. Se convierten dichos puntos en objetos hechos de mallas, posteriormente se toman una serie de fotos del objeto en diferentes ángulos y se procede a clasificarlo, identificando el porcentaje de asertividad que tiene el algoritmo al momento de definir el tipo de objetos 3D al que pertenece.

Palabras clave: Energía, Equipo electrico, Estructuras reticulares, Nubes de puntos, Objetos 3D, Sostenibilidad, Subestaciones eléctricas.

0. INTRODUCCIÓN

La creación de modelos tridimensionales es una práctica común en la actualidad para todo tipo de infraestructura. Para realizar el proceso de captura de información existente en activos construidos, se tienen opciones como los escaneos laser 3D y los levantamientos con fotogrametría que dan paso a los archivos de nubes de puntos.

Procesar estos archivos implica asignar recursos calificados y herramientas especializadas para realizar dicha labor, adicionalmente se requieren grandes cantidades de tiempo para realizar un modelo riguroso que no presente errores humanos, tan comunes en estos procesos.

Aunque para el sector de la arquitectura, ingeniería y construcción existen herramientas (software) que indican ayudar en el proceso de creación de modelos tridimensionales, a la conclusión que se ha llegado luego de evaluar las herramientas comerciales más mencionadas y disponibles, es que ninguna de ellas ofrece una solución para la infraestructura eléctrica.

Por lo anterior en este trabajo de investigación se desarrolló un aplicativo web, que usa los resultados de los escaneos LÍDAR para evaluar la probabilidad de que los puntos de los escaneos conforman determinado equipo eléctrico y en la misma iteración convierte esos puntos en un objeto 3D que puede usarse en la creación de modelos As-built.

La solución propuesta permite reducir los tiempos, la cantidad de veces que los recursos interactúan con las nubes de puntos evitando así errores humanos, presenta objetos 3D que representan la realidad de los equipos y adicionalmente no es necesario un alto desempeño computacional.

1. PLANTEAMIENTO DEL PROBLEMA

Modelo de Información de Construcción (BIM)

En la actualidad BIM se consolida en la industria de la arquitectura, ingeniería y construcción como una herramienta fundamental, esto se debe a que permite generar ahorros en los tiempos de ejecución de los proyectos, impactando los costos, la cantidad de recursos que intervienen en el proyecto y todos los demás ítems asociados al ciclo de vida. Por esta razón es tan importante la automatización, ya que va a permitir mejorar la productividad del sector de la construcción de la infraestructura, brindando una mirada colaborativa e integrada. “Los mercados relacionados con BIM, aunque presentan una desaceleración económica a nivel global, se estima que continuaran creciendo de forma estable esto de acuerdo con las estimaciones de Pietra Gartzzen [20]”

Utilizar BIM en las etapas previas de un proyecto en fases de levantamiento de condiciones existentes y también en fases de operación y mantenimiento, ha demostrado un potencial muy amplio en la mejora de la gestión de los activos [21] y en la toma de decisiones usando la información que tiene el modelo BIM. A pesar de las investigaciones y los avances, sigue siendo BIM un desafío, esta situación es lo que impulsa nuevas investigaciones en este campo.

Generación de Modelos BIM - Desafíos

En proyectos de infraestructura de cualquier tipo, en particular para este trabajo el interés es infraestructura eléctrica, se utiliza un escáner láser para identificar la infraestructura existente, los resultados del escaneo son nubes de puntos y a partir de estas se pretende crear un modelo llamado As-built; los datos que se obtienen de este proceso es un archivo con coordenadas (x,y,z) donde cada punto tiene una coordenada, de esta forma el archivo tiene un numero finito de puntos espaciales que representa un objeto. La información que arroja un escaneo, aunque es útil porque permite ver la infraestructura levantada una vez se ha dejado la obra (ver imagen 1), requiere que los datos que se obtienen sean procesados para desarrollar los objetos 3D, este proceso normalmente se realiza por personal calificado del área de dibujo o del área de ingeniería y se hace con software de casas reconocidas como Autodesk o Bentley; la opción que se escoja dependerá de muchos factores, entre ellos el país en que se realice el levantamiento, el tipo de infraestructura a levantar, los recursos que tiene el proyecto para ser desarrollado, etc. Para este caso se menciona solo una casa de software Autodesk, sin desconocer que existen otras que aportan beneficios iguales o más valiosos que la mencionada.

El proceso que se debe realizar para limpiar la nube de puntos es manual y requiere un tiempo considerable donde se presentan altas probabilidades de errores humanos lo que hace ineficiente, costoso y lento el proceso. Para cerrar esas brechas varias casas de software han creado herramientas que de manera semiautomática generan componentes, por ejemplo, Autodesk Plant 3D [22]. Aun así, no existe un software que sea 100% automático y capaz de generar los objetos que son tomados en el escaneo sin presentar limitaciones por el tipo de infraestructura, generando a partir de esa información el modelo As-built de alta calidad, esta situación es lo que permite que esta área de investigación este activa.

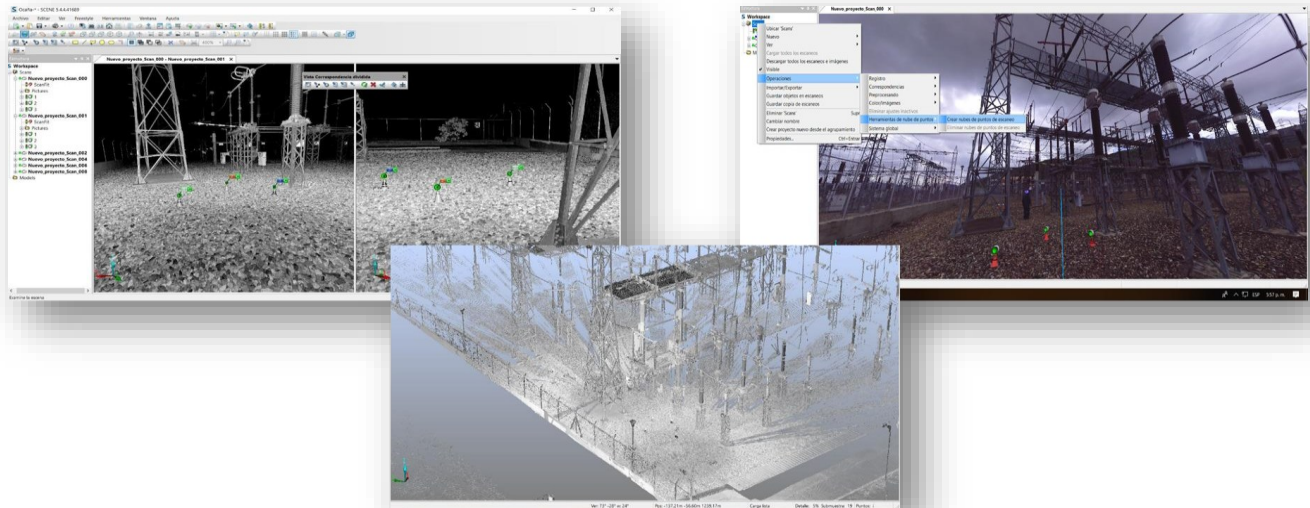


Fig. 1. Resultado de un levantamiento con nube de puntos.

Fuente: elaboración, propia con base en levantamiento realizados.

También en el mercado existe software comercial que promete realizar el procesamiento de nubes de puntos de manera automática, por esta razón se analizan los softwares disponibles en el mercado para validar si estos resuelven la necesidad que se tiene con las subestaciones eléctricas. En la siguiente tabla se presenta un resumen de la evaluación de los softwares.

| Software | Descripción | Promesa del software | Resultados obtenidos |
|----------------------|--|---|---|
| Undet for Revit [23] | Undet es una herramienta para Revit que mejora el escaneo. | Undet es una herramienta que promete optimizar el uso de nube de puntos en la creación de modelos digitales en Revit. Esta herramienta indica que | Se solicita a la casa fabricante del software un demo tipo plug-in que se instala en Revit. El plug-in solicita cargar las nubes de puntos en |

| Software | Descripción | Promesa del software | Resultados obtenidos |
|---------------------------------|--|--|---|
| | | <p>mejora la eficiencia y la gestión del tiempo.</p> | <p>formato .e57 y presenta la nube de puntos en el espacio tridimensional de trabajo.</p> <p>La herramienta carga como objetos 3D puertas y muros.</p> <p>La herramienta no logra convertir los objetos de subestaciones eléctricas en modelos 3D.</p> |
| <p>NeRF 3D [24]</p> | <p>Es una herramienta de inteligencia artificial para crear videos de gran realismo.</p> | <p>La herramienta realiza la captura de cuatro fotografías de un mismo objeto, persona o lugar; luego la herramienta combina esas capturas para crear un video de 360 grados.</p> | <p>Se solicita a la casa fabricante del software un demo, se instala el demo y los resultados que ofrece no aplican a subestaciones eléctricas.</p> |
| <p>CloudCompare [25]</p> | <p>Es una herramienta de procesamiento de nubes de puntos y mallas 3D.</p> | <p>CloudCompare es una herramienta que promete procesar y generar una visualización de grandes nubes de puntos 3D para convertirla en mallas, también puede generar otros tipos de formatos de geometría, la herramienta indica que recibe gran cantidad de datos y hace</p> | <p>Se evidencia que CloudCompare es una herramienta gratuita para la visualización, limpieza, segmentación y análisis de nubes de puntos.</p> <p>La capacidad de CloudCompare en la reconstrucción 3D es limitada cuando se trata de estructuras industriales con geometrías complejas.</p> |

| Software | Descripción | Promesa del software | Resultados obtenidos |
|--------------------------------------|--|--|--|
| | | <p>un avanzado análisis y segmentación.</p> | <p>Esta herramienta requiere software complementario como Blender, Revit, entre otros, para lograr una reconstrucción 3D precisa. Para esta investigación no resulta una opción viable por lo mencionado anteriormente.</p> |
| <p>Meshlab [26]</p> | <p>Es un software de código abierto que realiza el procesamiento y edición de mallas 3D.</p> | <p>Este software promete ser una herramienta avanzada para limpiar, segmentar, reconstruir y texturizar mallas resultado de una nube de puntos, este software es apto para impresiones 3D y modelado genérico.</p> | <p>Se solicita a la casa fabricante del software un demo, este software es útil para generar una malla base desde una nube de puntos compleja. El nivel de detalle y precisión que se requiere para los equipos eléctricos requiere un refinamiento posterior. Realizar este refinamiento implica un trabajo más extenso que requiere de otros softwares como Blender, Rhino, Revit o SolidWorks y no brinda una automatización.</p> |
| <p>Bentley Pointools [27]</p> | <p>Es un software que permite la animación, visualización, y la edición de nube de puntos.</p> | <p>Bentley Pointools, promete potenciar la velocidad de procesamiento de las nubes de puntos,</p> | <p>Esta solución es muy potente para el manejo de las nubes de puntos, pero no genera objetos 3D directamente desde las</p> |

| Software | Descripción | Promesa del software | Resultados obtenidos |
|--------------------------------------|--|--|---|
| | <p>Realiza la limpieza preparando de manera intuitiva las nubes de puntos para facilitar el uso de los resultados en aplicaciones posteriores.</p> | <p>presentando una visualización de alto rendimiento con una segmentación avanzada. Este software se comercializa como una solución profesional, con edición por capas, detección de interferencias y generación de animaciones.</p> | <p>nubes de puntos, por lo cual se debe completar con otro tipo de soluciones para generar un flujo de trabajo completo.</p> |
| <p>EdgeWise [28]</p> | <p>Es un software de modelado que automatiza los flujos de trabajo de escaneos BIM.</p> | <p>Este software promete acelerar el flujo de trabajo hasta en un 73%, modelando estructuras, tuberías, ductos, paredes y bandejas automáticamente. También promete automatizar la conversión de nubes de puntos en modelos BIM.</p> | <p>EdgeWise no genera mallas cerradas en formatos .obj/.stl; no modelo objetos complejos automáticamente. Adicionalmente es un software comercial bajo suscripción con precios desde €2.500 por año en Europa y en estados unidos su precio puede llegar a USD 10.000 por una licencia. Todo depende del mercado.</p> |
| <p>Trimble RealWorks [29]</p> | <p>Es un software de flujos de trabajo automatizados de nubes de puntos.</p> | <p>Este software promete ser una herramienta que importa los datos de escaneos, analiza los datos y modela parcialmente</p> | <p>Trimble RealWorks es un software muy amplio para registro, limpieza, generación de mallas y superficies, aun así, no responde cuando se usan</p> |

| Software | Descripción | Promesa del software | Resultados obtenidos |
|----------|-------------|----------------------------------|--|
| | | mallas, secciones y modelos CAD. | nubes de puntos de los objetos que se presentan en subestaciones eléctricas. |

Tabla 1. Resumen del software comercial.

Fuente: elaboración, propia.

Por las razones indicadas en la Tabla 1, se concluye que no existe un software comercial que permita dar una solución a la creación de modelos y objetos 3D a partir de nubes de puntos de manera semi automática o totalmente automática, los software mencionados aunque hacen uso de las nubes de puntos en procesos como limpieza, generación de mallas, edición por capas, segmentación y creación de animaciones no son una solución para el caso de subestaciones eléctricas.

Obtener un modelo As-built semi automático como resultado del escaneo 3D es una necesidad para optimizar los procesos y mejorar los resultados. En esta investigación se evaluaron cinco algoritmos de optimización considerados los más reconocidos en este campo, de acuerdo con el análisis realizado se seleccionó cual algoritmo se ajusta a los objetivos que se quieren lograr; se debe tener en cuenta que cada uno de estos algoritmos de manera independiente indica que puede crear objetos 3D a partir de nube de puntos. Para el análisis se tuvo en cuenta que los resultados fueran óptimos vs las limitaciones, de esta forma se seleccionó el algoritmo que ayudo a lograr el objetivo en la creación de objetos 3D a partir de un archivo de datos, en particular se logró identificar cuál es la calidad del objeto expresándola en porcentaje de asertividad.

Entonces es importante evaluar ¿De qué manera la aplicación de un algoritmo de optimización impacta la generación de un modelo As-built?, ya que no existe un software comercial que cumpla los requerimientos de creación de objetos 3D para subestaciones eléctricas, debido a que las soluciones actuales se enfocan en elementos arquitectónicos, como columnas, muros, losas y vigas. Dicho modelo tiene como fuente los escaneos 3D, para esto se considera que hay una relación directa entre ambos, algoritmo de optimización y precisión del modelo.

Se realizaron pruebas experimentales, en las que con el algoritmo seleccionado se comparó el modelo As-built generado para diferentes equipos. Ya que lo que se aborda es no solo el proceso de digitalización

de infraestructura construida, sino garantizar que los datos siguen un enfoque que va a permitir validar o rechazar la hipótesis presentada.

Por estas razones la investigación desarrollada es un avance en esta temática tan importante para la generación de modelos As-Built, es una propuesta estructurada que facilita la creación de objetos 3D usando como fuente nubes de puntos, todo esto para contribuir a la digitalización de las subestaciones eléctricas.

2. JUSTIFICACIÓN

El sector de la arquitectura, la ingeniería y la construcción enfrenta muchos retos, entre ellos la digitalización de la infraestructura construida, ya que gran parte de dicha infraestructura se encuentra en la etapa de renovación, sobre todo en el sector eléctrico donde los cambios son una constante. Esto se debe a que los intereses de Colombia, es ir aumentando las capacidades operativas del sistema de transmisión nacional [30], por esta razón, es imperiosa la labor de digitalizar los activos del sector eléctrico, bajo automatismos que ayuden en el proceso y faciliten la labor.

Aún con los desarrollos tecnológicos actuales para el sector, contar con modelos digitales [31], [32], [33], [34], [35], [36], [37], [38] que representen fielmente el estado actual de la infraestructura eléctrica continúa siendo un desafío, esto se debe a que se requiere invertir mucho tiempo en esta tarea y adicionalmente los recursos tecnológicos necesarios son costosos, sin dejar de lado que se requiere de personal calificado para realizar esta actividad. Esta situación está generando la necesidad de métodos eficientes e inteligentes para cerrar los vacíos entre la infraestructura física y la digital.

Es importante realizar una investigación sobre este tema ya que se va a validar y definir un sistema que automatice [39], [40], [41], [42], [43], [44] el proceso que existe entre el levantamiento de las nubes de puntos [45], [46], [47] y la creación del modelo 3D.

Desarrollar una herramienta que automatice beneficiará de manera directa los equipos de arquitectura, ingeniería, construcción y modeladores 3D, todos ellos se verán impactados por el aplicativo desarrollado; ya que estandariza el proceso de captura de información, la limpieza de las nubes de puntos, el procesamiento e interoperabilidad entre los formatos, dando por último como resultado un objeto 3D aplicable a modelos de infraestructura eléctrica. Con este trabajo se inicia el proceso de reconocimiento que puede servir para muchos más objetos, ya que la infraestructura eléctrica es mucho más amplia que lo que ha sido abordado en esta investigación.

En la actualidad se presentan muchas herramientas como las ya mencionadas que prometen impactar en los procesos de todos los sectores relacionados con infraestructura, entre ellas al sector eléctrico [48], más puntualmente a las subestaciones eléctricas [49], [50], [51] y particularmente a los equipos eléctricos de la subestación [52], [53], [54]. Sin embargo, un impacto real y fácilmente medible para la industria

son los tiempos que se asigna al personal (horas hombre) y la cantidad de veces que se debe repetir una actividad por vacíos en la información de entrada (número de versiones del entregable), por esta razón mejorar el reconocimiento de objetos y que sea una opción escalable a otras infraestructuras de los sectores de transporte [55], hidráulico y edificaciones públicas [56], [57], beneficiará no solo a estos sectores, sino también a la academia donde los procesos desarrollados con el estándar BIM ya empiezan a ser impartidos de manera complementaria con el diseño y modelado.

El aplicativo desarrollado es una propuesta al campo de investigación donde los algoritmos de segmentación semántica, redes neuronales, etc., continúan evaluando su alcance de aplicación, la calidad de la información que produce todo esto aplicado a diferentes activos [58], [59].

3. OBJETIVOS

3.1.GENERAL

Aplicar un algoritmo para la reconstrucción de estructuras reticulares que pueda ser usado en modelos BIM.

3.2.ESPECÍFICOS

- Investigar algoritmos existentes para la identificación de objetos en nubes de puntos.
- Implementar un prototipo utilizando una base de datos de nubes de puntos.
- Evaluar la precisión del objeto 3D expresada en porcentaje de asertividad, comparándola con el objeto real de la infraestructura eléctrica.

4. MARCO TEÓRICO O MARCO CONCEPTUAL

La investigación se desarrolló para crear objetos 3D que inician con las nubes de puntos. A continuación, se presentan conceptos, metodologías y tecnologías explicando cómo se realiza el procesamiento de los datos, para posteriormente usar esta información en los modelos BIM [60].

Es importante mencionar que obtener una infraestructura digitalizada que represente fielmente la realidad, es un avance que se ha dado en los últimos años. Tener esta infraestructura se debe al uso de escaneos laser terrestre y fotogrametría.

El resultado de utilizar las tecnologías antes mencionadas se evidencia en los archivos de nubes de puntos que es la fuente de los datos, con esta fuente de información se realizó el análisis, configuración del modelo y creación del entorno digital.

Para el sector de arquitectura, ingeniería y construcción, es fundamental contar con una nube de puntos cuando se requiere crear un modelo 3D de un activo construido que represente la realidad. Lo importante es que con base en la información que se encuentra en el archivo de nube de puntos se logra generar cada uno de los objetos que conforman el activo, incluso cada uno de los objetos puede ser parametrizado y posteriormente usarse en la creación de modelos BIM [61]. Todo esto para mejorar la precisión, la gestión y planificación de los proyectos.

Esta investigación se encuentra en un contexto tecnológico, que va dirigido a convertir los registros que se encuentran en una nube de puntos a un objeto 3D, pero usando un automatismo (algoritmo) que facilite el proceso. Para esto se debe tener en cuenta que el objeto debe contar con las siguientes características mínimas, como la geometría (largo, ancho y alto) y debe estar en formatos interoperables para que de manera posterior pueda usarse en un software BIM.

4.1.FUNDAMENTOS CONCEPTUALES

La investigación realizada requiere que sean comprendidos algunos conceptos que se encuentran asociados como lo son, el levantamiento del activo, procesamiento de nube de puntos y el modelado de objetos 3D.

4.1.1. Escaneo láser 3D y fotogrametría

Los escaneos láser 3D para funcionar y presentar resultados, inician enviando haces de luz para medir la distancia que existe entre el equipo y el objeto que está siendo medido, dicha medición se hace a gran velocidad, por el contrario, la fotogrametría digital lo que hace es usar imágenes superpuestas y a partir de la información presente en dichas imágenes reconstruye la superficie; hacer esto es posible gracias a los algoritmos de correlación. Estas técnicas permiten la generación de nubes de puntos donde la precisión dependerá del método usado [17].

4.1.2. Nubes de puntos

Las nubes de puntos son un grupo de coordenadas tridimensionales, este grupo de coordenadas representa una superficie de un objeto que hace parte de un entorno físico, cada una de estas coordenadas se representan por sus componentes (x,y,z) , adicionalmente las nubes de puntos incluyen atributos adicionales, colores , intensidad, reflectividad e imágenes asociadas, las nubes de puntos son el componente principal para generar modelos 3D [62], [63]

4.1.3. Procesamiento de nubes de puntos

“Puede formularse como un conjunto concreto de bloques de construcción parametrizados para lograr diferentes resultados. Por ejemplo, no existe diferencia algorítmica entre un algoritmo de detección de paredes, uno de puertas o uno de mesas; todos comparten el mismo bloque de construcción, que en este caso es un algoritmo de segmentación planar restringida. Lo que cambia en los casos mencionados es un subconjunto de los parámetros utilizados para ejecutar el algoritmo” [58].

Así el procesamiento de nube de puntos incluye varias etapas como el registro, filtrado de ruido, alineación de varios escaneos, clasificación de las geometrías, segmentación de las superficies y la generación de las mallas.

4.1.4. Objetos 3D y modelado paramétrico

Los objetos que habitan la realidad del mundo tridimensional son representados en el espacio digital como un objeto 3D, donde se preserva la forma, las dimensiones y las propiedades intrínsecas. Al generar un objeto 3D que nunca ha sido representado en un entorno digital, es suficiente con lograr dentro de la herramienta de modelado una identificación visual del objeto simplemente con mirarlo, pero cuando es necesario que el objeto cambie sus condiciones geométricas como alto, ancho, largo o en algunos casos presente un detalle mayor, es necesario para esto crear parámetros [64], luego a través de los parámetros se puede controlar cómo se comporta el objeto de acuerdo con la necesidad que se tenga de este. De esta manera un objeto o modelo paramétrico permite una gestión eficiente de la información y que pueda ser actualizado de una forma dinámica según las necesidades.

4.1.5. Modelo tridimensional

Un modelo 3D es la representación digital que se logra usando matemática y geometría, los modelos 3D pueden categorizarse como modelos geométricos, paramétrico o incluso semántico, el tipo de modelo depende del nivel de información que se tenga de los objetos que hacen parte de este [65]. Para esta investigación el modelo 3D se orientó a reconstruir los elementos que hacen parte de la infraestructura eléctrica (subestaciones) donde los datos vienen de nubes de puntos.

4.2. BASES TEÓRICAS Y METODOLÓGICAS

La geometría computacional y el modelo digital es fundamental en el proceso de crear objetos tridimensionales. Se parte desde datos discretos y se procede a realizar la reconstrucción geométrica; esta etapa se lleva a cabo identificando patrones, relaciones espaciales entre puntos cercanos y definiendo superficies continuas y mallas.

También es posible usar otras técnicas, donde se divide la nube de puntos en regiones más pequeñas y que guardan cierta uniformidad, esto se conoce como segmentación, lo que se logra es asociar la región con objetos ya conocidos para esto se utilizan algoritmos de detección de planos o clustering [16] para facilitar la identificación de los objetos.

Por último, el planteamiento Scan to BIM [47], se presenta como una “aplicación práctica que integra la captura de los datos con la generación de los objetos 3D”, todo esto se realiza para permitir una conexión entre la representación digital de la infraestructura y la realidad física.

Si se tiene una mirada metodológica a la creación de objetos 3D desde las nubes de puntos se requiere establecer un flujo de trabajo por etapas:

- ✓ Captura de datos
- ✓ Procesamiento de los puntos
- ✓ Modelado de los objetos
- ✓ Validación de los resultados.

Realizar este análisis implica el uso de herramientas especializadas que cumplan coherencia entre la geometría y el modelo.

4.3. ANTECEDENTES INVESTIGATIVOS

Proponer un algoritmo para el reconocimiento de objetos 3D donde la información de entrada proviene de escaneos con tecnología como LiDAR presenta múltiples desafíos, aun así, es un campo ampliamente investigado.

Por ejemplo, en el sector de movilidad autónoma es fundamental reconocer los objetos que se encuentran en un rango cercano, por esta razón este campo de investigación ha recibido fondos sustanciales para optimizar el proceso y presentar avances. La necesidad principal a la que deben responder los fabricantes de la tecnología que embarga los vehículos autónomos es garantizar la seguridad y que los datos que son analizados sean precisos en un entorno en constante cambio.

Por esta razón múltiples investigaciones se han enfocado en desarrollar técnicas para el procesamiento y optimización de los datos que se obtienen al usar tecnología LiDAR.

Una propuesta incluye algoritmos embebidos para lograr reducir los datos que toman los sensores, la propuesta fue realizada por D. Jung y D. Park [66] donde se utiliza detección semántica diferencial; todo esto para disminuir del orden del 80% la información que se transmite, manteniendo la confianza de los datos.

Otras investigaciones se enfocan en temas críticos como la confiabilidad de la información que está siendo procesada cuando se ve afectada por contaminantes como polvo, agua o aceite en los sensores, para este caso G. Jati et al. presenta “LIRAROC” que es un conjunto de datos realista, que analiza cada factor contaminante que afecta de manera importante el rendimiento de los sensores 3D [67]; de manera similar cuando se usa la visión por computadora con YOLOv4 según M. Sahal et al. [68] para una detección rápida, se puede demostrar el conocimiento que se tienen sobre las técnicas para vehículos autónomos donde el procesamiento de información se realiza en tiempo real.

Por ejemplo, J. Lambert et al. [69] han logrado realizar un análisis del desempeño de diez modelos de tecnología LiDAR 3D, dichas tecnologías son usadas de forma común en vehículos autónomos, donde se presentaron indicadores de rendimiento y se identifican problemas en rangos de distancias entre los 5 y los 180 metros; relacionadas con la exactitud y precisión de los equipos. Incluso en esta investigación se utilizó un peatón sintético (maniquí) y un vehículo dotado de la tecnología LiDAR 3D para de manera exhaustiva evaluar la conducción en vehículos autónomos.

Estas investigaciones y muchas más en el mismo campo evidencian el alto desarrollo tecnológico para los vehículos autónomos, que en la actualidad se presenta como uno de los beneficios de la transformación digital.

Para el caso de los vehículos la combinación de sensores, algoritmos, grandes bases de datos, bancos de pruebas estandarizadas y el análisis profundo de los datos, genera que el reconocimiento de objetos 3D sea preciso y seguro para el uso en la industria automotriz.

En cambio, para el sector de la arquitectura, ingeniería y la construcción, aunque se han presentado avances importantes con el mismo enfoque de reconocimiento de objetos 3D, los resultados continúan siendo limitados y divididos, en el caso de investigaciones como la de D. Laefer and L. Truong-Hong [6] en la cual se presentan automatismos para reconocer secciones de acero, usando librerías preconfiguradas y donde la información de entrada para el procesamiento proviene de nubes de puntos, los resultados logrados reconocen con exactitud más del 90% de las secciones de acero. Lo anterior es un avance importante, sin embargo, los desafíos como la irregularidad geométrica, la calidad de los datos, la pérdida de información y la intervención humana que aún es necesaria, ha limitado utilizar esta solución de manera general en proyectos estructurales.

En otra investigación como la de Z. Xiang et al [17], se usó segmentación semántica y fotogrametría inversa aplicando el algoritmo DeepLab, todo esto para identificar elementos estructurales como muros, columnas y losas para generar un modelo BIM As-built, donde la información de entrada proviene al igual que los casos anteriores de nubes de puntos tomada con tecnología LiDAR.

Esta orientación presentó mejores resultados en cuanto a la automatización, pero los desafíos persisten incluyendo la dependencia de imágenes de alta resolución y la limitación de aplicar el mismo enfoque a edificaciones con configuraciones complejas.

El panorama entre las tecnologías para vehículos autónomos donde se han desarrollado modelos generativos avanzados, como los presentados por Y. Song et al [12] donde incluso, para la reconstrucción y reducción de datos complejos se usa ecuaciones diferenciales estocásticas vs lo que está ocurriendo en el sector de la arquitectura, ingeniería y la construcción, donde la exploración de diversas tecnologías apenas comienza nos da un panorama general del alcance de las investigaciones.

Para el caso de los vehículos el reconocimiento 3D ha sido y está siendo ampliamente investigado, debido a las exigencias en seguridad y a la motivación de las marcas líderes en el mercado de presentar una solución infalible y ser los primeros en hacerlo, para de esta manera posicionarse ante los consumidores. En cambio, los avances aislados, soluciones imprecisas y automatizaciones deficientes que solo se pueden aplicar a edificaciones, justifican realizar más investigaciones en este campo que puedan ser aplicadas a todo tipo de infraestructura con el fin de cerrar las brechas presentes en el sector para generar modelos As Built.

El caso de investigación desarrollado en este trabajo pretende aportar una nueva propuesta para infraestructura eléctrica, que posteriormente en otras investigaciones pueda ser usada a que abarque todos los objetos que conforman una subestación eléctrica, incluso que el proceso usado sea de utilidad en otras infraestructuras o en otros casos de investigación futuros.

4.4.MARCO NORMATIVO Y ESTÁNDARES TECNOLÓGICOS

La gestión de la información y la interoperabilidad se apoya en diversos estándares internacionales que dan una orientación en el proceso de creación de los modelos 3D. La serie ISO 19650 “especifica los requisitos para la gestión de la información, en forma de un proceso de gestión, en el contexto de la fase de entrega de los activos y los intercambios de información dentro de dicha fase, usando BIM” [70], promueve la colaboración. La ISO 16739 define el formato IFC (Industry Foundation Classes) “intercambio de datos en la industria de la construcción y en la gestión de inmuebles” [71], este es un esquema de datos de estándar abierto que permite el intercambio de modelos digitales entre diferentes herramientas.

Para el caso de los equipos de escaneos se han generado formatos muy específicos para la gestión de los resultados, como el formato e57 que facilitan el uso de los resultados entre diferentes aplicaciones, estas soluciones tecnológicas permiten que puedan utilizarse en distintas herramientas los datos y de esta forma no se pierde integridad y precisión. En todo el proceso es prioritario usar estas normas para asegurar que los modelos que se construyen a partir de las nubes de puntos sean interoperables [64].

5. DISEÑO METODOLÓGICO O METODOLOGÍA

Los trabajos de investigación pueden presentar varios enfoques, entre ellos se encuentran el cualitativo donde el método de investigación busca comprender e interpretar un fenómeno para finalmente explicarlo, realizando un análisis de contexto. Este enfoque se base en información no numérica; los más comunes son entrevistas, registro, observaciones, entre otros. Lo que se logra con este enfoque es describir, patrones, relaciones y percepciones que aparecen naturalmente de acuerdo con el fenómeno estudiado.

Por otro lado, se encuentra el enfoque cuantitativo que se centra en la medición numérica de las variables, se realizan experimentos y mediciones directas, incluso se usa software que permita describir el fenómeno de una forma objetiva que pueda explicarlo e incluso predecirlo. Su fundamento es la captura de los datos y el análisis de ellos, donde se utilizan diversas herramientas para identificar correlaciones y patrones en las variables de estudio.

Este tipo de enfoque es el usado para el caso de investigación que se ha desarrollado en este trabajo, ya que presenta eficiencias mostrando que el método es preciso y que el aplicativo web tuvo el desempeño esperado.

5.1.ENFOQUE CUANTITATIVO

La investigación evaluó el desempeño de diferentes algoritmos, usando como información de entrada las nubes de puntos que fueron obtenidas de escaneos LiDAR realizado en subestaciones eléctricas, a partir de esa información se evaluó la precisión (porcentaje de asertividad) que se daba en la identificación de un equipo electrico recopilando el tiempo de ejecución del proceso.

5.2.PROCEDIMIENTO METODOLÓGICO

La investigación se llevó a cabo en varias etapas:

| Etapas de la investigación | Objetivo específico | Actividades |
|--|--|--|
| Análisis del proceso de captura de puntos, caracterización tecnológica y revisión bibliográfica. | Investigar algoritmos existentes para la identificación de objetos en nubes de puntos | Revisa la forma en la que se hace la captura de los datos en campo con equipos LÍDAR. |
| | | Seleccionar diferentes algoritmos que permitan usar las nubes de puntos para crear objetos 3D. |
| | | Búsqueda en la base de datos de la EAFIT como Science Direct, Scopus, IEEE, ASCE Library, análisis y selección de los documentos relevantes. |
| Selección tecnológica. | Implementar un prototipo utilizando una base de datos de nube de puntos | Se selecciono el algoritmo que se ajusta a la investigación, indicando la razón por la cual este se diferencia de los otros evaluados. |
| Aplicación práctica, pruebas, evaluación y documentación. | Evaluar la precisión del objeto 3D expresada en porcentaje de asertividad, comparándola con el objeto real de la infraestructura eléctrica | El algoritmo seleccionado se aplicó a un mismo caso práctico identificando las etapas necesarias para que presente resultados, las ventajas y limitaciones |
| | | Se realizo la validación del modelo generado por el algoritmo. |
| | | Redacción de los resultados, gráficos, hallazgos y recomendaciones |

Tabla 2. Resumen procesamiento metodológico.

Fuente: elaboración, propia.

En esta investigación el proceso de referenciación de las fuentes se hizo realizando una correcta citación, se utilizaron herramientas como Litmaps para realizar un correcto rastreo del origen de las fuentes, adicionalmente se realizó una descripción de cada parte el proceso, brindando transparencia en el uso de las herramientas para las cuales se cuenta con software licenciado. De esta manera se aseguró el rigor metodológico.

5.3.TIPO Y DISEÑO DE LA INVESTIGACIÓN

Para abordar esta investigación se aplicó un perfil descriptivo y tecnológico, porque se utilizaron conocimientos teóricos para presentar una solución en caminata a generar un objeto 3D y modelos 3D. La solución que se presento es practica y replicable a otros tipos de objetos.

De una manera descriptiva se ha realizado este proceso, teniendo en cuenta la forma en la que se hace la captura de los datos en campo con equipos LiDAR, se dio seguimiento al procesamiento de los datos, los criterios de creación del contenido (objeto 3D) y valoración de los resultados.

Posterior a esto se realizó un análisis del proceso y se presentaron opciones de mejora, evidenciando que es necesario un algoritmo que ayude con el procesamiento de las nubes de puntos y con la creación de los objetos. Se realizó el análisis de documentación existente en casos similares, de esta manera parte de ese conocimiento fue aplicado a modelos BIM [72] para subestaciones eléctricas.

Es necesario aclarar que en la metodología que se desarrolló no se usan variables, por el contrario, se revisan los procesos existentes, se desarrolló una alternativa y se registró su aplicabilidad, comportamiento y efectividad.

5.4.SOFTWARE DE ANÁLISIS

Aunque existen gran variedad de algoritmos de segmentación semántica, redes neuronales, redes neuronales convolucionales y modelos de aprendizaje profundo, para esta investigación se hizo un rastreo y se encontraron que cinco algoritmos (PointNet++, CNN, DeepLab, DCNN y EfficientNet) ya han sido usados en otros sectores como vehículos autónomos, edificaciones e identificación de imágenes; por esta razón se decide evaluar estos cinco que ya han sido configurados para realizar una clasificación y segmentación de objetos en nubes de puntos, también los algoritmos usados realizan una clasificación con imágenes y representaciones complejas pre entrenadas.

5.5.TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE INFORMACIÓN

La búsqueda de información se realizó usando dos estrategias:

La primera donde se revisó la información existente como modelos 3D, escaneos a patios de subestaciones eléctricas y para esto se usaron herramientas de modelado como Revit y ReCap; que son las herramientas usadas en la transformación de las nubes de puntos a modelos 3D.



Fig. 2. Modelo 3D – Revit, equipo electrico descargador.
Fuente: elaboración, propia con base en planos de fabricación.

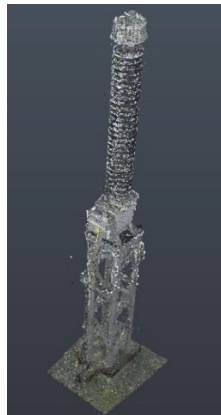


Fig. 3. Modelo 3D – Recap, equipo electrico descargador.
Fuente: elaboración, propia con base en los resultados de las nubes de puntos.

La segunda fue la búsqueda de documentación en bases de datos científica (IEEE, Scopus, ASCE Library, ScienceDirect, entre otras), estas fuentes se usaron para identificar los antecedentes que se han realizado en este campo, las teorías y métodos sobresalientes en el procesamiento de las nubes de puntos y su conversión a objetos.

Realizar estas búsquedas fue posible gracias a la fuente científica que tiene disponible la universidad EAFIT, dicha fuente permito identificar documentación precisa y valiosa.

Los filtros de búsqueda como palabras claves, años consultados y país fuente de la información se presentan a continuación:

| Base de datos | Ecuación de búsqueda |
|---------------|---|
| ScienceDirect | "point cloud" AND "3D OBJECT" |
| Scopus | TITLE-ABS-KEY (("POINT CLOUD") AND ("3D OBJECT")) AND (LIMIT-TO (PUBYEAR , 2024) OR LIMIT-TO (PUBYEAR , 2025)) AND (LIMIT-TO (AFFILCOUNTRY , "United States")) |
| IEEE Xplore | ("All Metadata": "point cloud") AND ("All Metadata": "3D object") |
| ASCE Library | "POINT CLOUD AND 3D OBJECT" |

Tabla 3. Ecuación de búsqueda.

Fuente: elaboración, propia.

| Base de datos | N° Artículos encontrados | Año de búsqueda | | N° Artículos seleccionados | Delimitación geográfica | Enlace de búsqueda |
|---------------|--------------------------|-----------------|------|----------------------------|-------------------------|---|
| | | 2024 | 2025 | | | |
| ScienceDirect | 108 | 59 | 49 | 13 | Estados Unidos | https://www-sciencedirect-com.ezproxy.eafit.edu.co/search?qs=%22point+cloud%22+AND+%223D+OBJECT%22&date=2024-2025&articleTypes=REV&lastSelectedFacet=articleTypes |
| Scopus | 115 | 58 | 57 | 22 | Estados Unidos | https://www-scopus-com.ezproxy.eafit.edu.co/results/results.uri?sort=plf-f&src=s&sid=fbc29537ced79f500b544ff150a737eb&sot=a&sdt=a&cluster=scoaffilctry%2C%22United+States%22%2Ct%2Bscopubyr%2C%222024%22%2Ct%2C%222025%22%2Ct&sl=55&s=TITLE-ABS-KEY+%28+%28+%22POINT+CLOUD%22+%29+AND+%28+%223D+OBJECT%22+%29+%29&origin=searchadvanced&editSaveSearch=&txGid=b06a04e1c0ee f848da2e059442680452&sessionSearchId=fbc29537ced79f500b544ff150a737eb&limit=10 |

| Base de datos | N° Artículos encontrados | Año de búsqueda | | N° Artículos seleccionados | Delimitación geográfica | Enlace de búsqueda |
|---------------|--------------------------|-----------------|------|----------------------------|-------------------------|---|
| | | 2024 | 2025 | | | |
| IEEE Xplore | 1266 | 682 | 584 | 39 | Estados Unidos | https://ieeexplore-ieee-org.ezproxy.eafit.edu.co/search/searchresult.jsp?action=search&matchBoolean=true&queryText=(%22All%20Metadata%22:point%20cloud)%20AND%20(%22All%20Metadata%22:3D%20object)&highlight=true&returnFacets=ALL&returnType=SEARCH&matchPublish=true&ranges=2024 2026 Year |
| ASCE Library | 577 | 363 | 214 | 45 | Estados Unidos | https://ascelibrary-org.ezproxy.eafit.edu.co/action/doSearch?AllField=POINT+CLOUD+AND+3D+OBJECT&AfterYear=2024&BeforeYear=2026&queryID=33%2F1072710258&startPage=&target=default&content=articlesChapters |

Tabla 4. Resumen resultados de búsqueda.

Fuente: elaboración, propia.

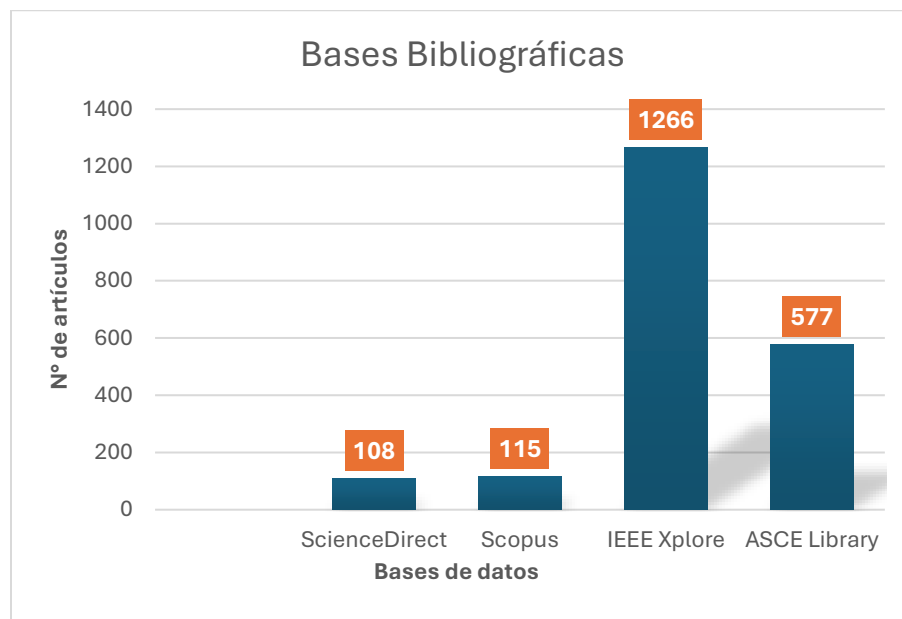


Fig. 4. Número de artículos encontrados en las bases bibliográficas.

Fuente: elaboración, propia.

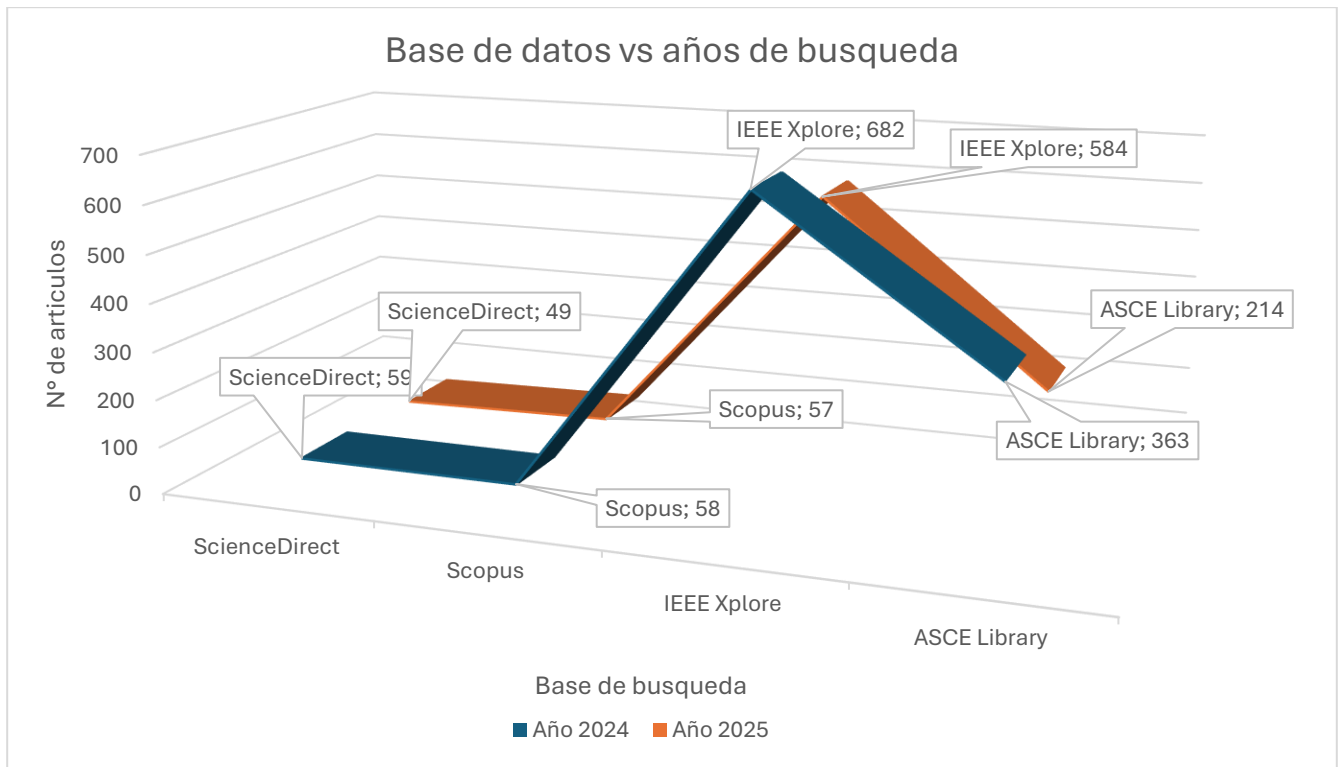


Fig. 5. Base de datos vs año de búsqueda

Fuente: elaboración, propia.

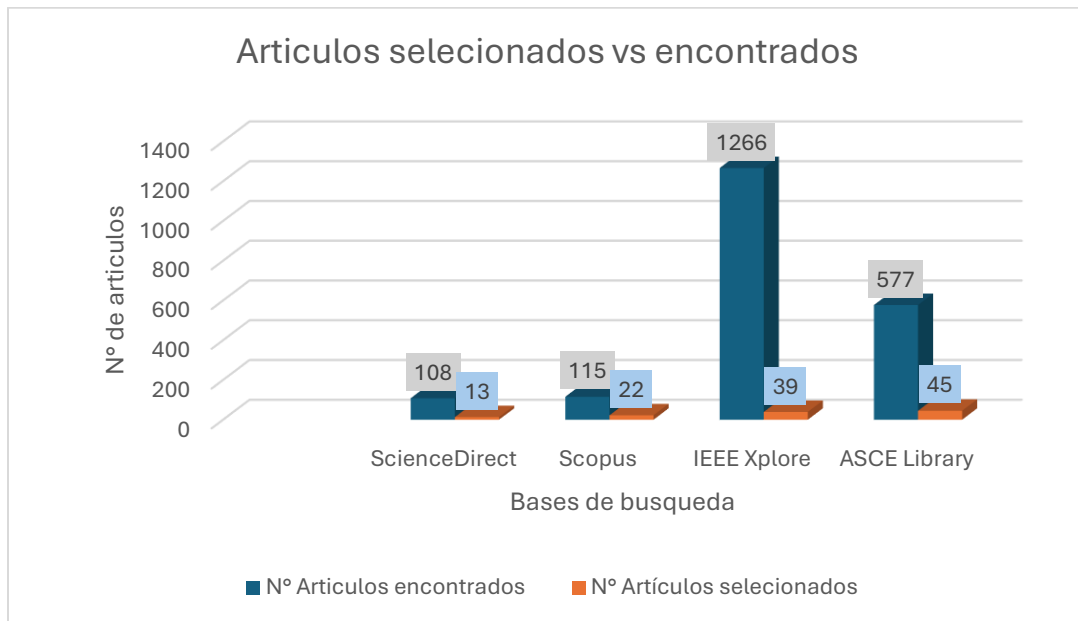


Fig. 6. Artículos seleccionados para revisión y análisis

Fuente: elaboración, propia.

5.6.CRITERIOS DE BUSQUEDA

Es importante aclarar los siguientes puntos:

- ✓ La búsqueda inicialmente abarco un periodo de un año (2024-2025), esto se debió a que se buscó la información más reciente, ya que el desarrollo y los métodos de análisis de los algoritmos cada día da pasos agigantados.
- ✓ Se limito la búsqueda solo a estados unidos, esto en vista que son ellos quienes más fuentes de información abierta y de libre consulta disponen, incluyendo comunidades de practica como GitHub.
- ✓ Una vez que se encontraron suficientes fuentes bibliográficas se amplió el periodo de búsqueda.
- ✓ Cuando se realizó la ampliación de la búsqueda se hizo utilizando la herramienta Litmaps, para acelerar la revisión de la literatura y realizar una búsqueda más exhaustiva ya que se puede usar la red de citas para encontrar las fuentes primarias de la información.
- ✓ Al usar Litmaps se incluyeron en las fuentes bibliográficas artículos anteriores al año 2024.

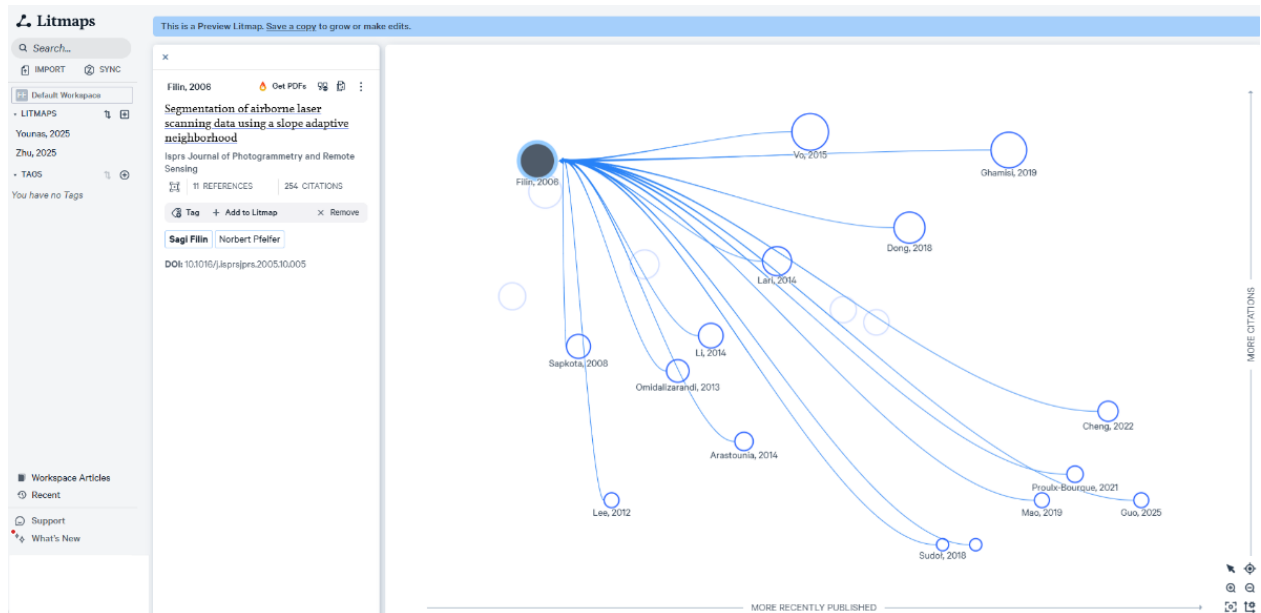


Fig. 7. Red de citas bibliografías.

Fuente: Litmaps.com.

- ✓ Una vez que se revisaron todas las fuentes bibliográficas seleccionadas en primera instancia, se conservaron ochenta y siete referencias que guardan una relación directa con la investigación.

5.7.VARIABLES DE ESTUDIO

Las variables de estudio para este trabajo de investigación fueron tres tipos:

| Tipo de variable | Nombre de la variable | Comentario |
|-------------------------|--|---|
| Independiente | Algoritmo aplicado | Algoritmo seleccionado, esta variable es la que se ajusta para identificar como es la creación de objetos 3D. |
| Dependiente | Porcentaje de asertividad | Concordancia entre el objeto real y lo detectado por el algoritmo, esta variable se expresa en porcentaje (%). |
| | Tempo de ejecución | Se refiere al tiempo transcurrido entre la entrada de los datos y el reporte del objeto listo para integrarlo en otras plataformas. Se expresa en segundos. |
| Control | Tamaño de nubes de puntos | Es el tamaño de los datos de escaneo, que conforman las nubes de puntos. El tamaño debe ser controlado para que el procesamiento sea optimo. |
| | Condiciones de realización del escaneo | Deben realizarse de manera que se mantenga condiciones como la estabilidad del dispositivo, distancia a los objetos escaneados, traslapos entre escaneos, es importante controlar estas condiciones para que no se vea afectada la calidad de la captura de los datos |

Tabla 5. Variables de estudio.

Fuente: elaboración, propia.

6. RESULTADOS

6.1. DESCRIPCIÓN DE LOS ALGORITMOS

A continuación, se describen los cinco algoritmos seleccionados para el procesamiento de nubes de puntos 3D:

En primer lugar, el algoritmo PointNet++ [73], [74], [75], [76]; segmentación semántica para nubes de puntos 3D, para este caso el algoritmo usa representaciones no estructuradas de un espacio en cuestión, este espacio está compuesto por puntos con coordenadas x,y,z , también el algoritmo puede opcionalmente usar atributos como color e intensidad de estos colores. Para este caso las nubes de puntos no tienen un orden fijo o en cualquier caso vecinos definidos. PointNet fue de las primeras arquitecturas que funcionó directamente sobre nubes de puntos, su principal éxito es que se logró respetar la inmutabilidad del orden de los puntos. Aun así, PointNet no captura correctamente las relaciones locales entre los puntos, de acuerdo con las falencias mencionadas, en su siguiente actualización paso a ser PointNet++ donde se extendió a una estructura jerárquica agrupada localmente, para evaluar información global y local al mismo tiempo.

Entre los puntos claves de PointNet++, se tiene la selección que realiza del muestreo de puntos para aplicar un muestreo del punto más lejano (FPS farthest point sampling) realiza esto para elegir un conjunto más pequeño que sea representativo, lo que permite es reducir la cantidad de puntos con los que se van a trabajar y para cada punto seleccionado toma vecinos cercanos para formar una area local.

Las principales ventajas de PointNet++ es captar las relaciones locales y globales de una forma jerárquica, presentando una plataforma más robusta para distribuciones no uniformes de densidad, generando una diferencia muy importante en comparación con PointNet inicial; razón por la cual se descartó para esta investigación PointNet original y se evaluó PointNet++. La desventaja es el costo computacional para conjunto de puntos grandes. Además, las operaciones de agrupamiento y muestreo generan un alto costo sobre todo en nubes con una alta densidad. Otra dificultad es el uso de este algoritmo es la interpolación y propagación de las características presentes en las nubes de puntos.

Se ha evidenciado que con métodos recientes se pueden mejorar aspectos como la precisión, velocidad y exactitud, aun así, el costo computacional continúa siendo alto, por estas razones se puede indicar que el rendimiento en términos generales de PointNet++ es muy reducido comparado con algoritmos modernos.

El segundo algoritmo CNN [77], [78]; redes neuronales convolucionales, donde una red neuronal es diseñada para que procese los datos que tiene una estructura de rejilla como imágenes o videos, para esto lo que usa son operaciones convolucionales, las principales ventajas del algoritmo CNN es que es capaz de compartir parámetros y usar vecindades locales, permitiendo detectar patrones con menores cantidades de datos ya que no se está usando una red completamente conectada. La aplicación de Kernels (filtros) ayuda en la creación de mapas con determinadas características aportando a la reducción de cómputo, esta es una de las principales ventajas. Además, se extraen características jerárquicas automáticamente, aunque entre las limitaciones se encuentra la reducción de resolución espacial, lo que obviamente dificultará la localización exacta en los procesos de segmentación de los puntos; adicionalmente se requieren muchos datos cuando es necesario realizar un entrenamiento desde cero [79].

El tercer algoritmo en análisis es DeepLab [80], [81], [82]; es un grupo de modelos de aprendizaje profundo, donde DeepLab es la segmentación arquitectónica semántica de imágenes, este algoritmo está basado en redes neuronales convolucionales profundas (CNN) presentando mejoras donde lo que se busca es mantener la resolución espacial y capturar contexto en varias escalas, una de las ventajas de este algoritmo es que maneja un buen equilibrio entre el contexto de las nubes de puntos y la resolución espacial, también se ha mejorado su eficiencia mediante el uso de convoluciones separadas. Aunque las mejoras realizadas en DeepLab son notorias, la integración de estos refinamientos es costosa y en tareas de segmentación 3D o dominios en los que no se presentan imágenes, este algoritmo no es directamente aplicable y se deben realizar adaptaciones, sus principales aplicaciones es en vehículos autónomos, imágenes de órganos y tejidos, incluso es usado en imágenes de satélites.

El cuarto algoritmo en análisis es DCNN [83], [84]; redes neuronales convolucionales profundas por sus siglas en inglés (Deep Convolutional Neural Network) este también se puede entender como CNNs con varias capas, donde aprende a través de representaciones jerárquicas, también se suele normalizar por lotes usando redes ya entrenadas cuando se tienen pocos datos, entre sus principales ventajas esta la

capacidad de representar funciones complejas, tiene resultados buenos con las tareas visuales, es decir, es capaz de clasificar, detectar y segmentar entre diferentes dominios.

Por último, se evaluó el algoritmo EfficientNet [85]; este es una familia de modelos de redes neuronales convolucionales (CNN), este algoritmo es propuesto por Google en el año 2019; en primera instancia se desarrolló para tareas de imágenes de computadora clasificando dichas imágenes; la principal característica es que puede alcanzar una precisión alta usando menos parámetros y generando un menor costo en cuanto al uso computacional, en comparación con los algoritmos mencionados anteriormente. A partir del primer análisis realizado con EfficientNet las versiones de este algoritmo se han enunciado con B0, este algoritmo es un modelo base más pequeño y con el paso del tiempo se han realizado versiones más grandes desde la versión B0 hasta la versión B7.

Para lograr que EfficientNet-B0 funcione se requiere un tipo de bloque convolucional, este funciona usando MobileNetV2 que tiene una estructura de botella invertida, en primer lugar, expande los canales, luego aplica la convolución y por último reduce los canales, esto incluye conexiones residuales lo que lo hace eficiente en el proceso computacional. Una de las innovaciones es que en vez de escalar solo la profundidad, también puede hacerlo con el ancho o la resolución, es decir que EfficientNet-B0 escala las tres dimensiones al mismo tiempo y realiza esto de una forma equilibrada. Este algoritmo es diseñado para que mediante un proceso de búsqueda automática se pueda obtener resultados mejorando los rendimientos y mantener un costo computacional racional.

| N° | Categoría | Método | Objetivo | Adquisición de datos |
|----|---------------------------------|---|---|---|
| 1 | Segmentación 3D. | PointNet++, segmentación semántica para nubes de puntos 3D. | Clasificación y segmentación de objetos en nubes de puntos 3D no estructuradas. | Escaneos LiDAR, sensores RGB-D, datos tridimensionales. |
| 2 | Visión clásica por computadora. | CNN, análisis y perspectivas a fondo. | Clasificación de imágenes, detección de objetos y red neuronal diseñada para | Escaneos LiDAR, datos tridimensionales, ImageNet. |

| N° | Categoría | Método | Objetivo | Adquisición de datos |
|----|--------------------------|---|--|---|
| | | | que procese los datos que tienen una estructura de rejilla como imágenes o videos. | |
| 3 | Segmentación semántica. | DeepLab, inmersión profunda en el procesamiento visual avanzado. | Delimitación precisa de objetos en imágenes (segmentación semántica), resolución espacial y capturar contexto en varias escalas. | Escaneos LiDAR, datos tridimensionales. |
| 4 | Redes profundas. | DCNN, redes neuronales convolucionales profundas. | Extraer representaciones complejas de imágenes para clasificación o detección, se usan redes ya entrenadas cuando se tienen pocos datos. | Escaneos LiDAR, datos tridimensionales, ImageNet. |
| 5 | Clasificación eficiente. | EfficientNet-B0, modelo base de redes neuronales convolucionales. | Clasificación de imágenes 3D de forma eficiente y precisa, puede alcanzar una precisión alta usando menos parámetros. | Escaneos LiDAR, datos tridimensionales, ImageNet. |

Tabla 6. Algoritmos evaluados.

Fuente: elaboración, propia.

A partir del análisis anterior donde se describieron los cinco algoritmos para el procesamiento de nubes de puntos 3D, se presenta a continuación, cual fue el algoritmo que se usó para el caso práctico.

6.2. SELECCIÓN DEL ALGORITMO

En el caso de PointNet++, que es una arquitectura calificada para el procesamiento de nube de puntos 3D, este algoritmo no proyecta el uso de imágenes, por esta razón sería la fuente más fidedigna ya que se usan los datos tridimensionales, aun así, la aplicación de este algoritmo presenta limitaciones para la tarea que se requiere; entre ellas se encuentra.

- ✓ Tiene una dependencia de gran cantidad de datos y todos ellos deben estar etiquetados, esto es necesario para el entrenamiento ya que requiere realizar miles de ejemplos de cada clase que se va a evaluar, condición que no se cumple con la información disponible.
- ✓ Las nubes de puntos tomadas en un escaneo 3D, generan un archivo de extensión e57, este archivo generalmente presenta irregularidades, vacíos, ruido de objetos o personas que quedaron referenciados al momento del escaneo, todo esto afecta la estabilidad del modelo.
- ✓ Por último, si se realiza el entrenamiento usando PointNet++ se requiere una unidad de procesamiento gráfico (GPU) de alto rendimiento y asignar un mayor tiempo en el procesamiento de los datos.

Por lo anterior, aunque PointNet++ es ideal para cuando se tienen datos procesados con anterioridad, adicionalmente se puede asegurar que los datos 3D están limpios y que la cantidad de datos que cumple las características mencionadas es suficiente, este algoritmo sería una opción, pero para la investigación no se cumplen con estas condiciones razón por la que se descarta PointNet++.

Siguiendo con redes convolucionales clásicas o (CNN), las cuales presentaron los primeros avances en el campo visual computacional, lo que permitiría para la investigación contar con más información fidedigna de ejemplos realizados en el pasado, aun así este algoritmo presenta una estructura fija e ineficiente que limita la capacidad de generación comparándolo con modelos modernos, en términos generales las CNN tiene menor precisión y eficiencia cuando se procesan imágenes de alta resolución, cuando los datos son reducidos hay mayor tendencia de sobre ajuste lo que genera errores, por las razones

indicadas los CNN son considerados como un modelo base para referencias académicas, pero se descarta para su uso en arquitecturas principales.

En el caso de DeepLab la arquitectura de este algoritmo está orientada a la segmentación semántica, donde el principal propósito es definir etiquetas a cada píxel en una imagen, ofreciendo buenos resultados para identificar componentes específicos, pero se queda corto al momento de clasificar globalmente objetos complejos. Para nuestro caso implementar DeepLab implicaría hacer una segmentación de las partes de los equipos (aisladores, cuchillas, soportes, cajas, estructura), esto no solo se sale del alcance planteado, sino que dicha segregación no presentaría una solución eficiente del problema, donde se quiere pasar de las partes (puntos) al todo (objeto 3D), también la complejidad computacional que se requiere es mucho mayor.

Las redes neuronales convolucionales profundas (DCNN) permiten ajustar la cantidad de filtros y capas, esto permite tener flexibilidad en la experimentación de los resultados, aunque las DCNN carecen de una optimización automática y en cuanto al respaldo documental de este tipo de arquitecturas gran parte es empírico, generando riesgos de sobre ajuste cuando se tienen un número de datos reducido, adicionalmente se requiere mayor tiempo para que el algoritmo alcance resultados competitivos, ya que no se cuenta con modelos anteriormente entrenados.

EfficientNet es un algoritmo que por su capacidad de responder a la variabilidad de los datos y tener un menor requerimiento del tamaño de las nubes de puntos para arrojar resultados óptimos es de gran ayuda, también es un algoritmo que ha sido probado e incluso garantiza la estabilidad y repetitividad, presentando un rendimiento superior.

De acuerdo con el análisis se concluye que EfficientNet-B0 es el algoritmo que permite realizar las tareas de clasificación de puntos 3D en coordenadas (x, y, z), superando los otros cuatro algoritmos en las tareas de clasificación, tiene equilibrio entre la precisión del objeto, la eficiencia computacional y el tamaño del modelo que analiza. Este algoritmo ya tiene una arquitectura optimizada, se puede ejecutar en computadores con recursos limitados como equipos de escritorio o portátiles.

EfficientNet-B0 es un algoritmo moderno, probado en muchas ocasiones, es escalable para diferentes tamaños de nubes de puntos. Por lo anterior EfficientNet es la arquitectura seleccionada en comparación

con las alternativas evaluadas que se descartan por sus limitaciones relacionados con el tamaño de los dataset, la variabilidad del entrenamiento y su aplicabilidad práctica.

6.3.ETAPAS DE APLICACIÓN DEL ALGORITMO

La investigación se llevó a cabo en varias etapas, la primera de ellas fue convertir de manera automática los resultados que arroja los escáneres 3D (LiDAR) [86] que es el formato e57, este archivo es nativo para esta tecnología. El formato e57 es un archivo de nube de puntos muy completo donde se incluyen las coordenadas, colores, intensidad, reflectividad e imágenes asociadas. El archivo se convierte a formato x,y,z, que también es usado en la recopilación de datos de nube de puntos, pero mucho más simple ya que solo guarda la posición y el color de cada punto.

En una segunda etapa se convirtió el formato x,y,z obtenido en la etapa uno, a un formato .obj este es un modelo creado a partir de mallas 3D, que son las caras y aristas que conforman la superficie del objeto evaluado.

Por último, en la tercera etapa se creó el script que es la parte donde se utilizó el formato x,y,z u .obj para el entrenamiento del algoritmo, el mejor resultado se almacena en el archivo. pth y se usó este cerebro entrenado para hacer la clasificación de los objetos.

Este script también es la parte donde se dio un entorno visual o de aplicación como se muestra en la figura a continuación.

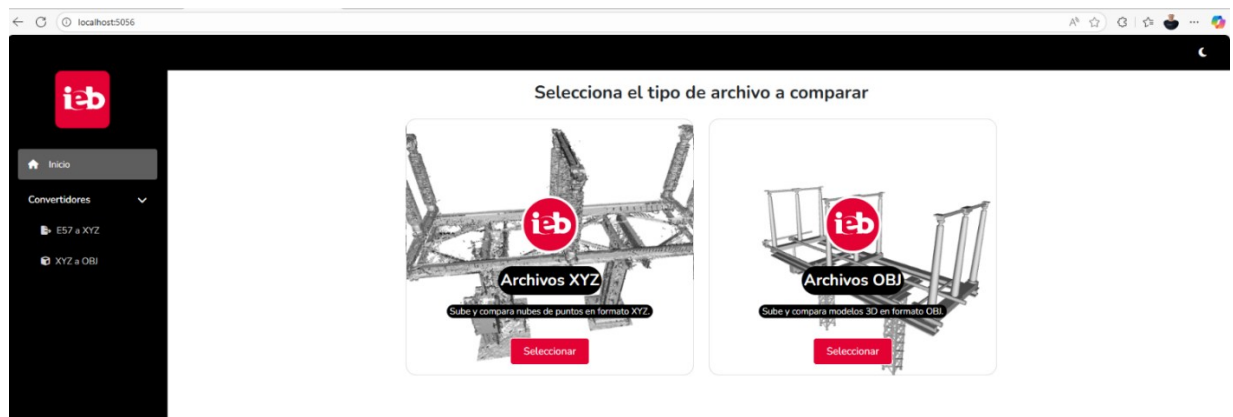


Fig. 8. Aplicativo web.

Fuente: elaboración, propia.

6.3.1. CONVERTOR E57 A XYZ

Descripción general

La aplicación web es la que permite convertir los archivos E57 que es el formato estándar en que se generan las nubes de puntos 3D, a formatos x,y,z que es un formato más simple de trabajar en coordenadas cartesianas. Esta aplicación es construida en Flask y utiliza Websockets para comunicarse en tiempo real.

Arquitectura

Backend (*app.py*)

- ✓ **Framework:** Flask + SocketIO
- ✓ **Librería Principal:** pye57 esta librería es para la lectura de los archivos E57
- ✓ **Modo Asíncrono:** eventlet es para el manejo de las operaciones asíncronas

Frontend

- ✓ **Tecnologías:** HTML5, JavaScript, TailwindCSS
- ✓ **Comunicación:** Socket.IO para las actualizaciones en tiempo real
- ✓ **UI:** Interfaz responsive (interfaz de usuario) con animaciones y feedback visual

Proceso de conversión

a) Recepción del archivo

```
@app.route('/upload', methods=['POST'])
def upload_file():
    file_id = str(uuid.uuid4())
    input_path = os.path.join(UPLOAD_FOLDER, f"{file_id}.e57")
    output_path = os.path.join(OUTPUT_FOLDER, f"{file_id}.xyz")
```

- ✓ Genera un ID único para el archivo
- ✓ Valida que el formato efectivamente es E57
- ✓ Guarda temporalmente en el directorio de uploads

b) Lectura del archivo E57

```
e57 = E57(input_path)
```

```
data = e57.read_scan_raw(0, ignore_unsupported_fields=True)
```

- ✓ Utiliza pye57 para leer el archivo
- ✓ `read_scan_raw`: Lee los datos sin transformaciones de pose
- ✓ Ignora los campos que no son soportados para obtener una mayor compatibilidad

c) Extracción de coordenadas

```
x = data.get("cartesianX", [])
```

```
y = data.get("cartesianY", [])
```

```
z = data.get("cartesianZ", [])
```

- ✓ Extrae las coordenadas cartesianas (X, Y, Z)
- ✓ Valida la integridad de los datos:
 - Mismo número de puntos en cada eje
 - Existencia de las coordenadas

d) Generación del archivo XYZ

```
with open(output_path, "w") as f:
```

```
    for i, (xi, yi, zi) in enumerate(zip(x, y, z)):
```

```
        f.write(f"{xi} {yi} {zi}\n")
```

- ✓ Formato: X Y Z por línea
- ✓ Procesamiento realizado por lotes para optimizar la memoria
- ✓ Actualización del progreso de cada `batch_size` de puntos

e) Sistema de progreso

- ✓ 10%: Carga inicial del archivo
- ✓ 30%: Lectura de datos del escaneo
- ✓ 50%: Extracción de coordenadas

- ✓ 70-95%: Generación del archivo XYZ
- ✓ 100%: Proceso completado

f) Manejo de errores

```
try:  
    # Operación  
except Exception as e:  
    emit('progress', {  
        'file_id': file_id,  
        'progress': 0,  
        'message': f" ✘ Error: {str(e)}"  
    })
```

- ✓ Captura y realiza la notificación de los errores en tiempo real
- ✓ Logging detallado para debugging
- ✓ Limpieza de los archivos en caso de un error

g) Limpieza de archivos

```
@app.route('/cleanup/<file_id>', methods=['POST'])  
def cleanup_files(file_id):  
    # Mantiene solo el último archivo procesado  
    files_to_delete = upload_files[:-1] + output_files[:-1]
```

- ✓ Elimina los archivos temporales después de la descarga
- ✓ Mantiene solo el archivo más reciente
- ✓ Previene la acumulación de archivos en servidor

h) Optimizaciones

1. Procesamiento por lotes

```
batch_size = max(total_points // 20, 1000)
```

- ✓ Equilibra el uso de la memoria y las actualizaciones UI (interfaz de usuario)

2. Manejo de memoria

- ✓ Lectura eficiente de los archivos E57
- ✓ Liberación de los recursos usados después del proceso

3. Comunicación asincrónica

- ✓ WebSockets para updates en tiempo real
- ✓ Reconexión automática en caso de desconexión

i) Formato de Salida (XYZ)

X1 Y1 Z1

X2 Y2 Z2

...

Xn Yn Zn

- ✓ Formato de texto plano
- ✓ Separado por espacios
- ✓ Un punto por cada línea
- ✓ Coordenadas en unidades originales del archivo E57

j) Requisitos del sistema

- ✓ Python 3.7+
- ✓ Flask
- ✓ pye57
- ✓ eventlet
- ✓ Espacio en disco para archivos temporales

k) limitaciones

- ✓ Realiza el procesamiento de un escaneo a la vez

- ✓ Mantiene solo el último archivo procesado
- ✓ No preserva información anterior

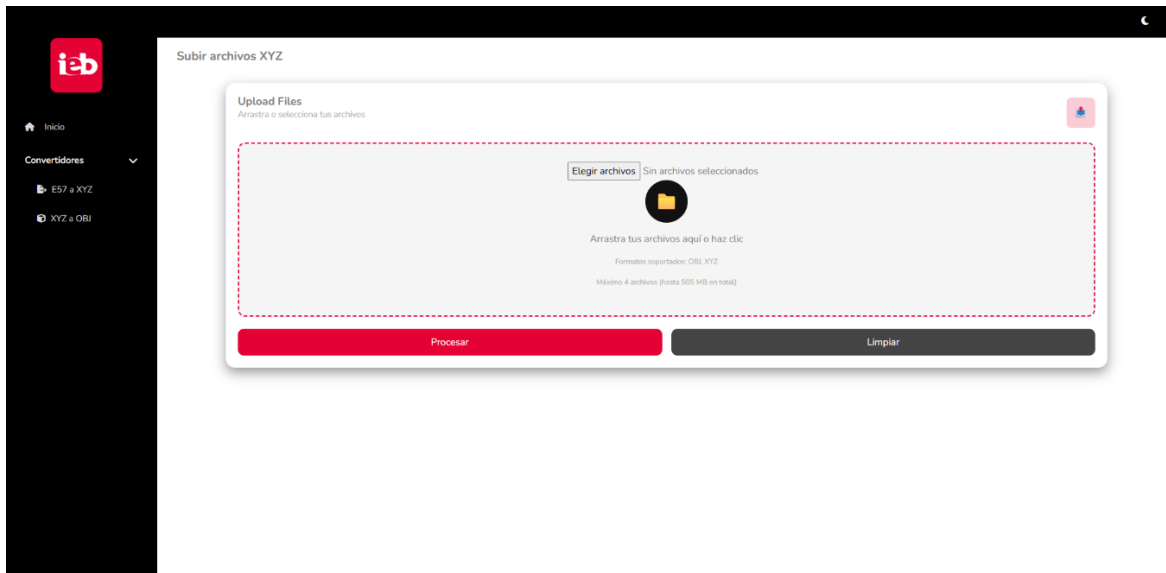


Fig. 9. Aplicativo web, carga de archivos formato e57.

Fuente: elaboración, propia.

6.3.2. CONVERTIDOR XYZ A MALLA 3D

1. Arquitectura del sistema

a) Componentes principales

- ✓ Frontend: HTML5, JavaScript, TailwindCSS
- ✓ Backend: Flask, SocketIO, Open3D
- ✓ Procesamiento: NumPy, Open3D
- ✓ Almacenamiento: En el sistema de los archivos en local

b) Dependencias principales

```
from gevent import monkey
from flask import Flask, request, send_file, jsonify
from flask_socketio import SocketIO, emit
import open3d as o3d
import numpy as np
```

2. Flujo de procesamiento detallado

a. Inicialización del servidor

```
app = Flask(__name__, static_url_path='/static', static_folder='static')
socketio = SocketIO(app,
    async_mode='gevent',
    ping_timeout=120,
    ping_interval=30,
    cors_allowed_origins="*",
    max_http_buffer_size=1e8)
```

b. Proceso de carga (upload_file)

○ Validación inicial

```
if not file.filename.lower().endswith('.xyz'):
    return jsonify({'error': 'Formato de archivo inválido'})
```

- ✓ Verifica la extensión .xyz
- ✓ Valida que el formato de salida sea (obj/ply/stl)
- ✓ Genera UUID único

○ Almacenamiento

```
file_id = str(uuid.uuid4())
input_path = os.path.join(UPLOAD_FOLDER, f"{file_id}.xyz")
file.save(input_path)
```

c. Procesamiento principal (process_file)

○ Carga de datos

```
points = np.loadtxt(input_path, usecols=(0, 1, 2))
```

- ✓ Lee las coordenadas XYZ

- ✓ Verifica que los valores sean válidos
- ✓ Maneja un submuestreo si son > 5M puntos (M=millones)

- **Creación de nube de puntos**

```
pcd = o3d.geometry.PointCloud()  
pcd.points = o3d.utility.Vector3dVector(points)
```

- **Estimación de normales**

```
pcd.estimate_normals(  
    search_param=o3d.geometry.KDTreeSearchParamKNN(knn=20),  
    fast_normal_computation=True)  
pcd.orient_normals_consistent_tangt_plane(k=20)
```

- ✓ Usa un KNN con k=20
- ✓ Optimizado el proceso para mejorar el rendimiento
- ✓ Orientación consistente de normales

- **Reconstrucción**

```
mesh, densities = o3d.geometry.TriangleMesh.create_from_point_cloud_poisson(pcd,  
    depth=10)
```

- ✓ Profundidad = 10 para el balance del detalle vs rendimiento
- ✓ Retorna la malla y las densidades para la limpieza

- **Post-procesamiento**

```
densities = np.asarray(densities)  
vertices_to_remove = densities < np.quantile(densities, 0.01)  
mesh.remove_vertices_by_mask(vertices_to_remove)
```

- ✓ Elimina los outliers basado en las densidades
- ✓ Recalcula las normales finales
- ✓ Optimiza la malla

d. Sistema de notificaciones

- ✓ Uso de SocketIO para las actualizaciones en tiempo real

- ✓ Los eventos de progreso se muestran con porcentajes
- ✓ Se presentan mensajes informativos personalizados

e. Gestión de archivos

```
@app.route('/cleanup/<file_id>')
def cleanup(file_id):
    # Mantiene solo archivos del último proceso
    for file in os.listdir(UPLOAD_FOLDER):
        if not file.startswith(file_id):
            os.remove(file_path)
```

3. Parámetros y configuraciones

a. Límites del sistema

- ✓ Máximo puntos que permite el proceso: 5,000,000
- ✓ Timeout Socket: 120s
- ✓ Buffer HTTP: 100MB

b. Parámetros de reconstrucción

- ✓ KNN normales: 20 vecinos
- ✓ Profundidad Poisson: 10
- ✓ Umbral densidad: 1%

4. Manejo de errores

a. Validaciones principales

```
if len(points) < 3:
    raise ValueError("Muy pocos puntos")
if not np.all(np.isfinite(points)):
    raise ValueError("Valores no válidos")
```

b. Recuperación de errores

- ✓ Reconexión automática del Socket
- ✓ Limpieza de los archivos temporales
- ✓ Logs detallados de los errores

5. Optimizaciones implementadas

a. Rendimiento

- ✓ Submuestreo inteligente
- ✓ Cálculo rápido de las normales
- ✓ Procesamiento asíncrono

b. Memoria

- ✓ Limpieza automática de los archivos
- ✓ Gestión eficiente del buffer
- ✓ Liberación proactiva de los recursos

6. Endpoints API

a. Rutas principales

- ✓ */upload*: POST para la subida de los archivos
- ✓ */download/<file_id>*: GET para la descarga
- ✓ */cleanup/<file_id>*: GET para la limpieza
- ✓ */check_status/<file_id>*: GET para el estado

7. Consideraciones de seguridad

- ✓ Validación de tipos de archivo
- ✓ Limpieza automática de archivos
- ✓ Tamaño máximo de archivo
- ✓ Permisos de directorio controlados

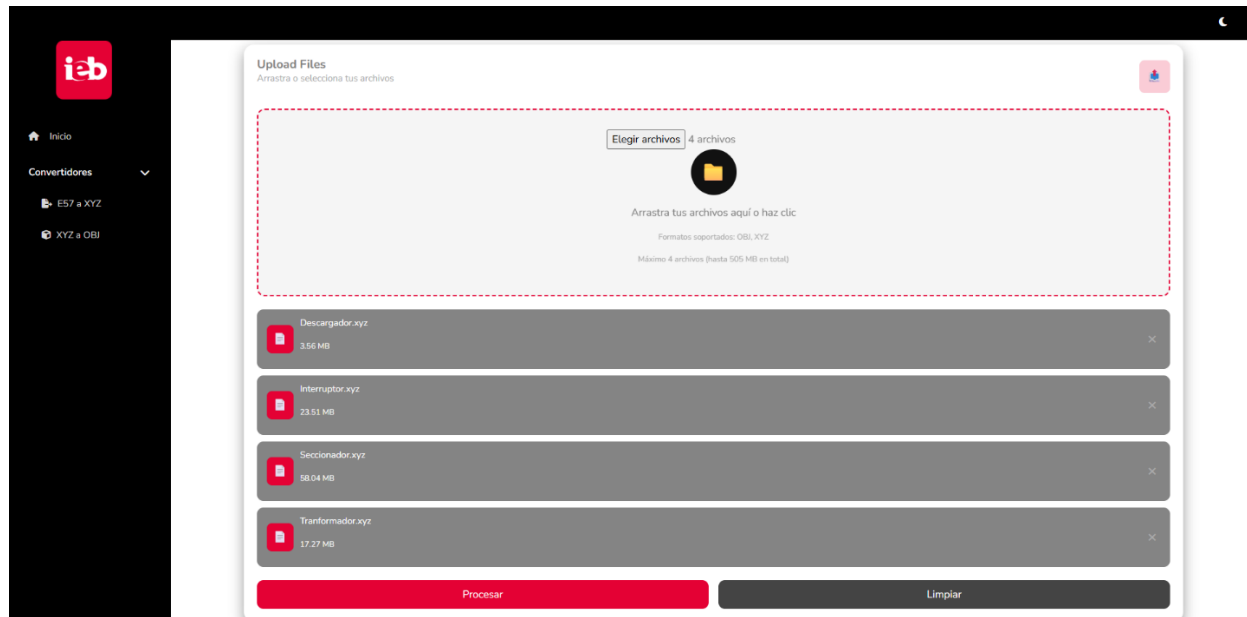


Fig. 10. Aplicativo web, procesamiento de los archivos a mallas 3D

Fuente: elaboración, propia.

6.3.3. DOCUMENTACIÓN DEL SCRIPT

A continuación, se implementa un sistema para clasificar los equipos eléctricos que tienen una vista 3D a partir de modelos en formatos .xyz o .obj. Para esto se utilizan redes neuronales convolucionales (CNN) con el algoritmo EfficientNet-B0, este último como backbone y entrenamiento supervisado con imágenes generadas con múltiples vistas.

El flujo general del script es:

1. Entrada: Modelos 3D (.obj) o nubes de puntos (.xyz).
2. Conversión/Normalización: Ajuste de escala, centro y colores.
3. Generación de vistas: Se capturan imágenes desde múltiples ángulos.

4. Entrenamiento: Si no existe un modelo previo entrenado.
5. Clasificación: Se clasifican los objetos 3D generados.
6. Resultados: Los objetos generados son guardados en `classification_results.json`.

El script crea y usa las siguientes carpetas de forma automática:

| Carpeta | Uso |
|------------------------------------|---|
| input_nube_puntos | Entrada de nubes de puntos <code>.xyz</code> . |
| input_objjs | Entrada de modelos <code>.obj</code> . |
| converted_rfa_objjs | Modelos <code>.obj</code> de referencia organizados por categoría para entrenamiento. |
| captured_images | Imágenes generadas desde vistas múltiples. |
| trained_models | Modelos entrenados guardados (<code>.pth</code>). |
| classification_results.json | Archivo JSON con resultados de clasificación. |

Tabla 7. Carpetas creadas por el script.

Fuente: elaboración, propia.

Categorías

Las categorías predefinidas son:

1. Descargador de sobretensiones,
2. Interruptor monopolar,
3. Seccionador trifásico,
4. Transformador de corriente y
5. Otro.

La categoría otro se usa para casos no identificados y estos no participan del entrenamiento.

Componentes principales

a. Selección del tipo de entrada

`select_input_type()`

Permite elegir entre:

- ✓ Nubes de puntos (`.xyz`)
- ✓ Modelos 3D (`.obj`)

b. Carga y normalización de la geometría

load_geometry(file_path)

- ✓ Centra el objeto en el origen
- ✓ Escala de forma robusta usando el percentil 95
- ✓ Aplica un color uniforme si no existe
- ✓ Elimina outliers y calcula normales

c. Generación de vistas múltiples

generate_multi_view_images(geometry, prefix, num_views)

- ✓ Captura imágenes desde varios ángulos (NUM_VISTAS=36 por defecto)
- ✓ Usa Open3D para renderizar el modelo en distintos puntos de cámara
- ✓ Guarda las imágenes en *captura_images*

d. Red neuronal

class EquipmentClassifier(nn.Module)

- ✓ Basada en EfficientNet-B0
- ✓ Se congelan capas iniciales para aprovechar el aprendizaje previo
- ✓ Clasificador final adaptado a las 5 categorías

e. Dataset personalizado

class equipmentDataset(Dataset)

- ✓ Gestiona el cargado de imágenes y etiquetas
- ✓ Aplica transformaciones de aumento de datos en entrenamiento:
 - Rotaciones aleatorias
 - Inversión horizontal/vertical
 - Jitter de color
 - Escala y normalización

f. Preparación de datos

prepare_training_data()

- ✓ Recorre las carpetas de referencia
- ✓ Genera vistas y etiquetas para cada categoría

- ✓ Devuelve lista de rutas de imágenes y sus etiquetas

g. Entrenamiento

train_classifier(imagen_paths, labels)

- ✓ División en 80% entrenamiento y 20% validación.
- ✓ *BATCH_SIZE* = 16
- ✓ *EPOCHS* = 50
- ✓ Optimizador: AdamW
- ✓ Programador de LR: ReduceLROnPlateau
- ✓ Guarda el mejor modelo en *trained_models/best_classifier.pth*.

h. Clasificación

classify_equipment(model, image_paths, top_k=3)

- ✓ Promedia las predicciones de todas las vistas.
- ✓ Devuelve la categoría más probable, la confianza y el top-3 de predicciones.

i. Ejecución principal

main()

- ✓ Pide tipo de entrada.
- ✓ Si no existe modelo, entrena uno nuevo.
- ✓ Carga el modelo entrenado.
- ✓ Clasifica todos los archivos de la carpeta de entrada.
- ✓ Guarda los resultados en *classification_results.json*.

Entrenamiento: épocas, porcentaje y validación

- **Épocas (EPOCHS = 50):** Número de veces que la red ve todo el dataset durante el entrenamiento.
- **Porcentaje de datos:**
 - **80% entrenamiento:** ajusta los pesos del modelo.
 - **20% validación:** Evalúa el modelo sin ajustar pesos para medir generalización.
- El modelo evalúa en validación tras cada época y guarda el **mejor** según la precisión.

- **ReduceLRonPlateau** reduce la tasa de aprendizaje si la pérdida de validación no mejora en 5 épocas.

Archivo.pth

- **Ubicación:** *trained_models/best_classifier.pth*
- **Contenido:**
 - Pesos entrenados (*model_state_dict*)
 - Estado del optimizador (*optimizer_state_dict*)
 - Precisión alcanzada (*best_accuracy*)
 - Época en la que se guardó (*epoch*)
- **Se carga con:**
 - `checkpoint = torch.load("trained_models/best_classifier.pth", map_location=device)`
`model.load_state_dict(checkpoint['model_state_dict'])`

Salida Final

Archivo *classification_results.json* con estructura:

```
{
  "archivo.obj": {
    "clasificacion_principal": {
      "categoria": "transformador",
      "confianza": 92.35
    },
    "top_predicciones": [
      {"categoria": "transformador", "probabilidad": 92.35},
      {"categoria": "interruptor", "probabilidad": 5.67},
      {"categoria": "seccionador", "probabilidad": 1.98}
    ]
  }
}
```

Diagrama de flujo del script *comparar.py*

flowchart TD

A[Inicio] --> B[Seleccionar tipo de entrada]

B -->|1| C[input_nube_puntos xyz]

B -->|2| D[input_objs obj]

C --> E[Verificar modelo entrenado.pth]

D --> E

E -->|No existe modelo| F[Preparar datos de entrenamiento]

F --> G[Generar vistas múltiples de modelos de referencia]

G --> H[Entrenar clasificador 80% train / 20% val, 50 épocas]

H --> I[Guardar mejor modelo en trained_models/best_classifier.pth]

E -->|Sí existe modelo| J[Cargar modelo entrenado.pth]

I --> J

J --> K[Leer archivos de entrada]

K --> L[Cargar geometría 3D]

L --> M[Generar vistas múltiples del modelo]

M --> N[Clasificar con el modelo entrenado]

N --> O[Guardar resultados en classification_results.json]

O --> P[Mostrar resumen por categoría]

P --> Q[Fin]

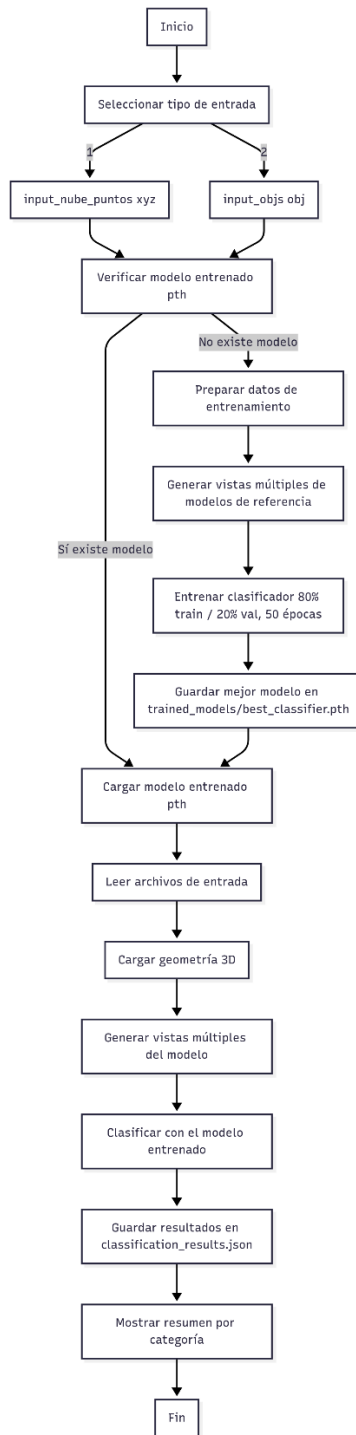


Fig. 11. Diagrama de flujo del Script.

Fuente: Online FlowChart [87].

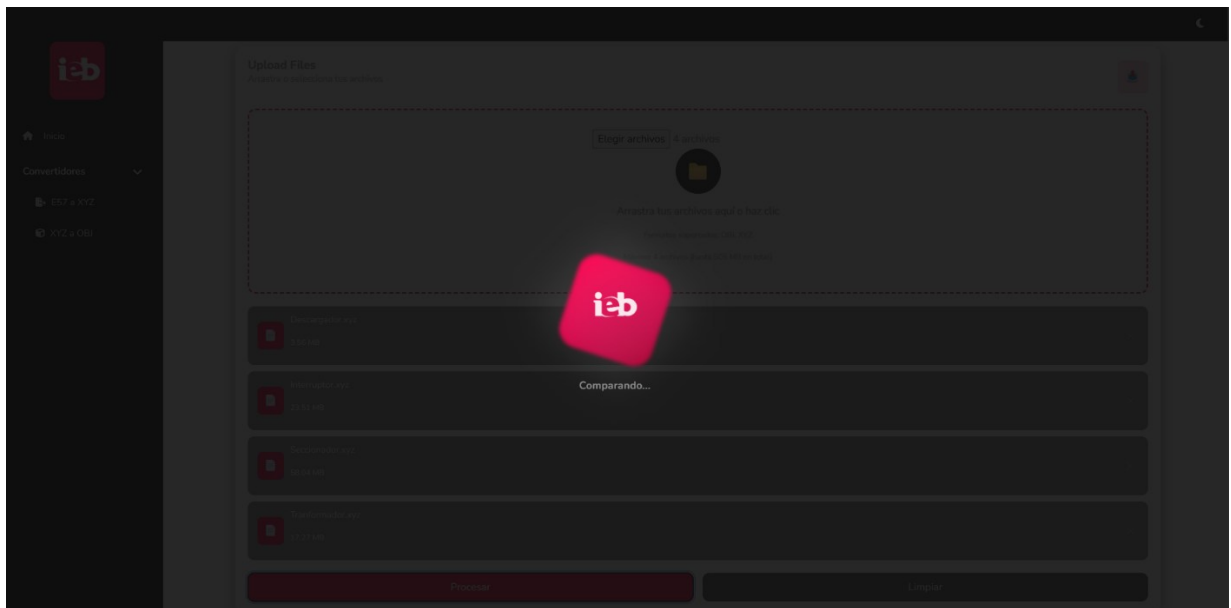


Fig. 12. Aplicativo web, procesamiento del script

Fuente: elaboración, propia.

6.4.RESUMEN DE RESULTADOS

Una vez definido el algoritmo que es EfficientNet-B0 y definidas las nubes de puntos sobre los cuales se realizará el procesamiento de los datos, se tomaron cinco equipos eléctricos:

1. Descargador de sobretensiones,
2. Interruptor monopolar,
3. Seccionador trifásico y
4. Transformador de corriente.
5. Otro.

La categoría “otro” aunque no es un equipo eléctrico se usa para casos no identificados y estos no participaron del entrenamiento.

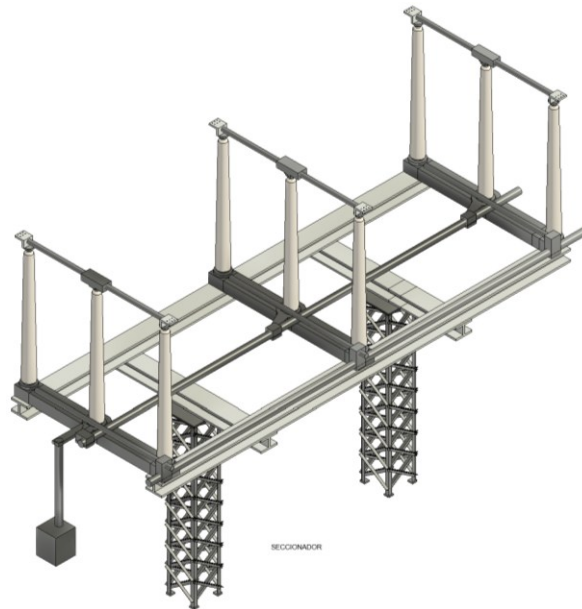


Fig. 13. Modelo 3D – Revit, equipo eléctrico seccionador.
Fuente: elaboración, propia con base en planos de fabricación.

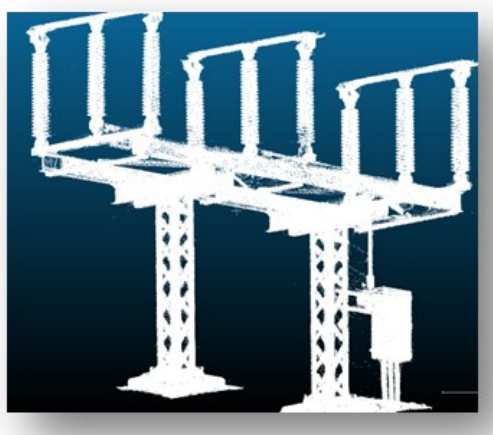


Fig. 14. Modelo 3D – Nube de puntos, equipo eléctrico seccionador.
Fuente: elaboración, propia con base en levantamiento realizados.

Para el entrenamiento del algoritmo, se utilizaron las nubes de puntos de los equipos (modelo 3D) como una maqueta digital, el algoritmo tomó treinta y seis fotos desde distintas posiciones, a cada foto se le preguntó de las opciones disponibles que tipo de equipo cree que tiene. ¿Esto es un interruptor?, ¿Esto es un transformador?, ¿Esto es un descargador?, etc. De esta manera se definen las clasificaciones.

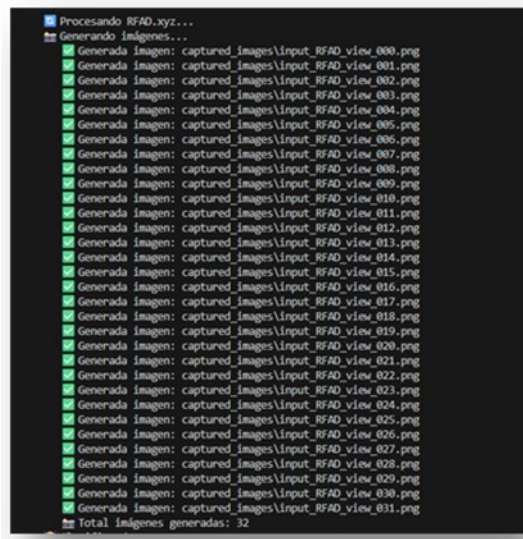


Fig. 15. Procesamiento del algoritmo al generar las imágenes.
Fuente: elaboración, propia con base en los análisis realizados.

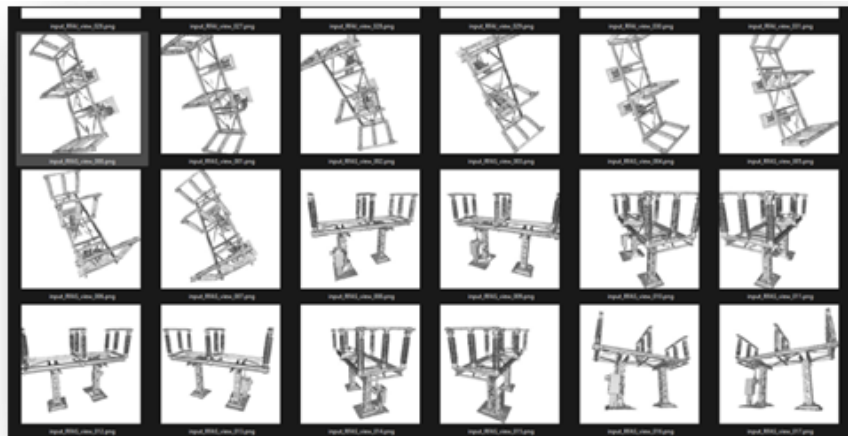


Fig. 16. Captura de imágenes - entrenamiento del algoritmo, equipo seccionador.
Fuente: elaboración, propia con base en los análisis realizados.

El programa inicia sin saber nada del objeto que está identificando, empieza a revisar las fotos, indica el nombre del equipo al que cree que pertenece y valida si efectivamente acertó o se equivocó, esta validación la hizo dejando en el espacio de trabajo las carpetas con el nombre de cada uno de los equipos en los cuales se está entrenando. Cuando el entrenamiento está listo y se le pregunta a que categoría pertenece toma el modelo 3D completo que es su información base, donde valida que todos los puntos pertenecen a un mismo equipo y de esta forma se atreve a categorizarse en un equipo determinado.

Cuando se equivoca realiza los ajustes para no cometer en la siguiente iteración el mismo error, este proceso se repitió como mínimo nueve mil veces, de esta manera aprende a reconocer los equipos eléctricos de la subestación.

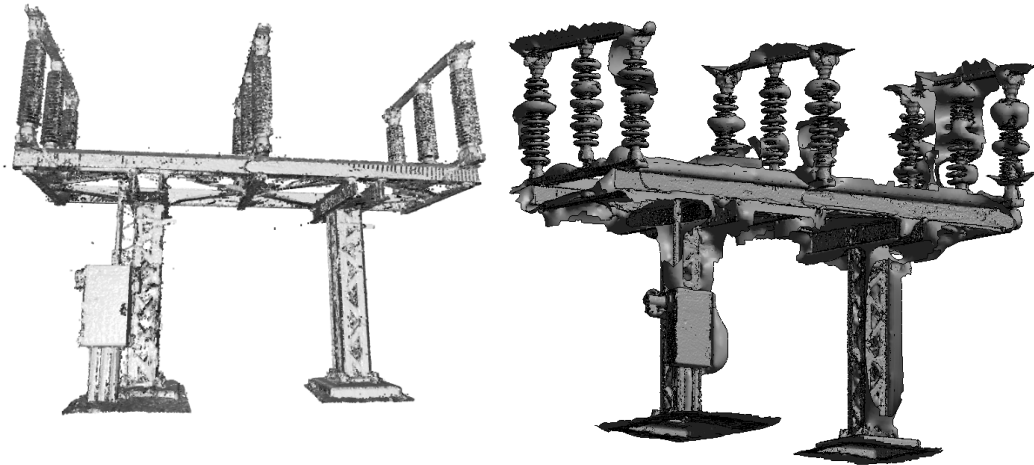


Fig. 17. Reconstrucción del equipo eléctrico, seccionador.

Fuente: elaboración, propia con base en los resultados obtenidos.

```
File Edit Selection View Go Run Terminal Help
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
classification_results.json
C: > Users > alexandra.moncada > OneDrive - INGENIERIA ESPECIALIZADA > Trabajo de grado MAESTRIA -EAFIT > classification_results.json > {} Transformador.xyz > {} clasificacion_principal
41 },
42 "Seccionador.xyz": {
43   "clasificacion_principal": {
44     "categoria": "seccionador",
45     "confianza": 99.75
46   },
47   "top_predicciones": [
48     {
49       "categoria": "seccionador",
50       "probabilidad": 99.75
51     },
52     {
53       "categoria": "otro",
54       "probabilidad": 0.08
55     },
56     {
57       "categoria": "transformador",
58       "probabilidad": 0.08
59     }
60   ]
61 },
```

Fig. 18. Resultados probabilísticos - confianza 99.75%, seccionador.

Fuente: elaboración, propia con base en los resultados obtenidos.

Para hacer esto se usó la red neuronal EfficientNet-B0. Esta red neuronal ya reconoce cosas genéricas, porque ha sido entrenada con millones de fotos de diversos objetos, lo que se hizo fue cambiar la parte

en la que aprende de una determinada categoría para que lo haga sobre la categoría en cuestión, para este caso es equipos eléctricos; para realizar este aprendizaje se utilizó un método llamado AdamW que lo que hace es ser el entrenador personal de la red neuronal donde le dice cambia esto, no cambies demasiado esta parte, cambia solo un poco esta parte; si la red neuronal no presenta mejores resultados, lo que se hace es que se reduce la velocidad de aprendizaje para que pueda afinar cada uno de los detalles.

Este proceso de aprendizaje se hizo por épocas, como se tienen treinta y seis fotos desde diferentes ángulos y se definieron cincuenta épocas, la red neuronal va a revisar las treinta y seis fotos cada una cincuenta veces y cada vez que ve todas las fotos se dice que se completó una época. Cuantas más épocas más aprende. Aun así, no se puede exceder la cantidad de épocas ya que es posible que memorice las fotos y no aprenda realmente a reconocer cosas nuevas, esto último es lo que se llama sobre ajuste.

Una vez que se le enseñó a la red neuronal, es necesario probar si realmente aprendió a identificar el equipo. Para esto, el 80% de las fotos se usaron para enseñarle a la red neuronal en la etapa de entrenamiento y el 20% de las fotos se guardaron para el momento de la validación. Esto permitió que cuando se realizó la validación se usaron fotos del equipo que la red neuronal nunca había visto. Este proceso permitió validar si la red neuronal realmente entiende o si solo está repitiendo lo que ha visto.

Por esto es importante para este análisis utilizar modelos entrenados, porque de esta manera la red neuronal ya cuenta con una experiencia adquirida, así cuando se ajusta al caso de análisis él no sabe nada de lo que está analizando. Posteriormente, cuando se realizó el entrenamiento específico para los equipos eléctricos aprendió a reconocerlos. El archivo que el algoritmo generó con esos resultados se guarda en un archivo con extensión .pth, Este archivo contiene toda la memoria y conocimientos adquiridos, de esta manera no tiene que volver a realizar el proceso de aprender desde cero cada vez que se ejecute.

Los resultados que emitió el algoritmo y que son del interés de este trabajo, son porcentajes, los cuales indican que tan seguro se encuentra que esa nube de puntos con la cual inicio el proceso y que convirtió en un modelo 3D, corresponde a la categoría de uno de los cinco equipos que están en estudio.

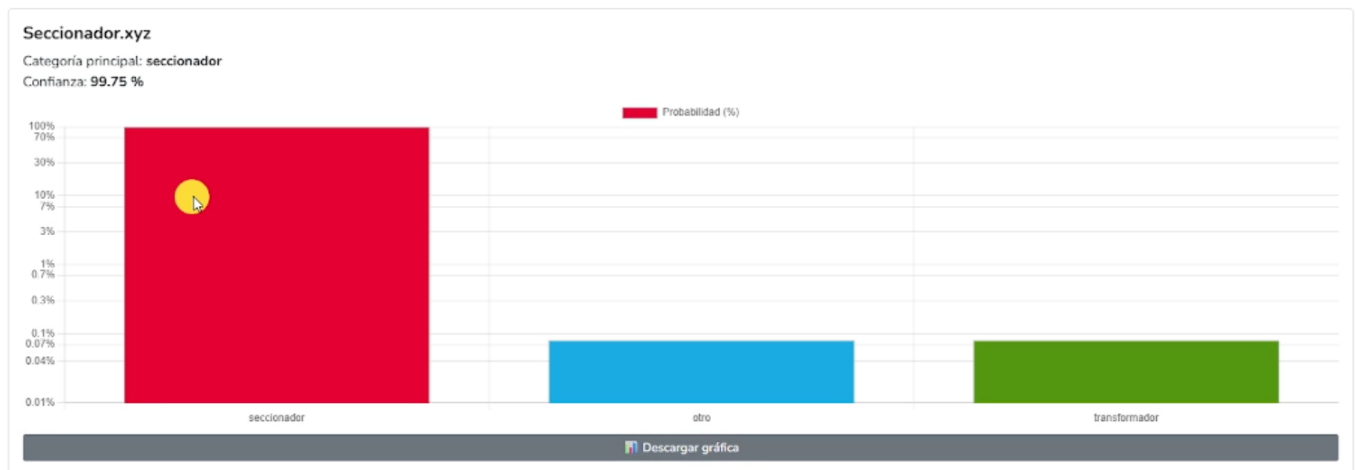


Fig. 19. Resultados gráficos - confianza 99.75%, seccionador.

Fuente: elaboración, propia con base en los resultados obtenidos.

El interés de obtener resultados expresados en porcentaje es que estos permitieron definir de una manera muy rápida la categoría a la que pertenecen los puntos del escaneo y el objeto 3D que ha generado, ya que la principal función que se le dió a este aplicativo fue ayudar en la creación de modelos 3D de infraestructura eléctrica existente. De esta forma el algoritmo ayuda a los equipos de modelación e ingeniería a crear modelo As-built que representan la realidad del activo construido con un porcentaje de calidad alto y a partir de formatos de nubes de puntos que vienen directamente de campo.

Los resultados presentados muestran que el sistema desarrollado ofrece una solución automatizada para los modelos de información del activo, aun así, se presentan desafíos, entre ellos el tamaño de las nubes de puntos que impacta directamente en la capacidad de procesamiento del algoritmo, información de nubes de puntos incompletas en gran parte de los casos esto se debe aún error humano, ya que es muy importante la capacidad del operario que realiza el levantamiento.

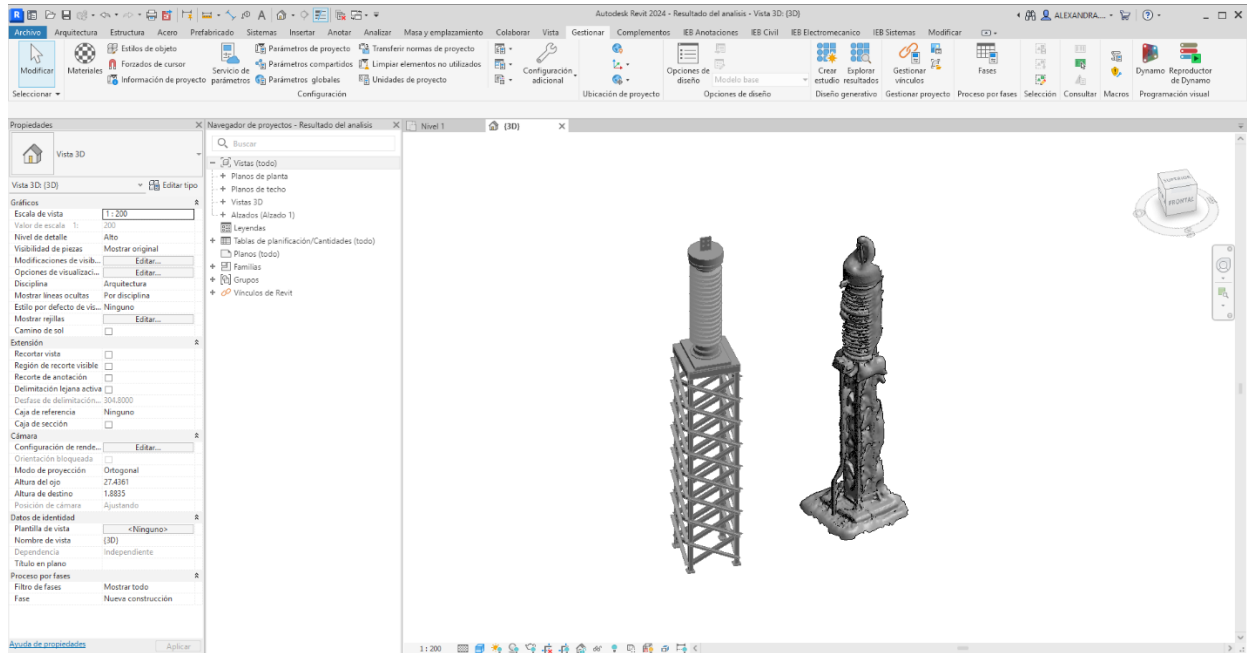


Fig. 20. Modelo 3D – Revit, equipo eléctrico descargador.

Fuente: elaboración, propia. Equipo modelado vs equipo reconstruido con el algoritmo.

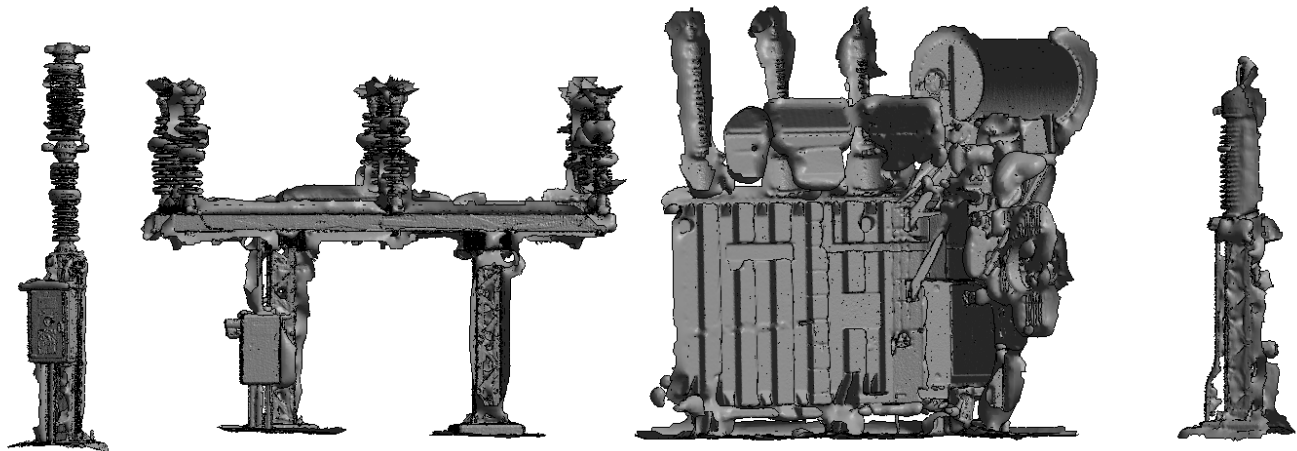


Fig. 21. Modelo 3D – Revit, equipos eléctricos.

Fuente: elaboración, propia. Equipos reconstruidos con el algoritmo.

7. CONCLUSIONES

El sistema propuesto mostró un alto rendimiento en cuanto al tiempo de desarrollo y la confianza de los objetos que evaluó, mostrando que este campo donde la generación automática de modelos As-built en la que la fuente de información proviene de una nube de puntos es prometedor, aun así, existen limitaciones entre ellas:

El aplicativo se enfoca actualmente en equipos eléctricos como descargador de sobretensiones, interruptor monopolar, seccionador trifásico y transformador de corriente. No se incluye en el alcance otros objetos que hacen parte de las subestaciones como estructuras de pórticos, bancos de ductos, cables, entre otros. Alcanzar la totalidad de los equipos eléctricos que se encuentran en el patio de una subestación hace parte de las siguientes fases de desarrollo.

La opción escogida para ayudar en el procesamiento y entrenamiento del aplicativo fue EfficientNet-B0, aunque ofrece buenos resultados, su efectividad puede variar cuando se aplique a otros datos o tipos de nubes de puntos.

La precisión del aplicativo depende en gran parte de la cantidad de imágenes que usa para entrenarse, para la investigación con un número de treinta y seis imágenes se obtuvieron resultados satisfactorios; sin embargo, para objetos con un nivel de complejidad mayor será necesario aumentar la cantidad de imágenes que hacen parte del entrenamiento.

Este proceso de generación automática de objetos 3D a partir de nube de puntos puede ser costoso en cuanto a los equipos de cómputo que requiere, aun así, con los resultados obtenidos y presentados en esta investigación se puede concluir que esta solución contribuye ya que desarrolla una estrategia innovadora para generar modelos As-built.

Aplicar un algoritmo de optimización impacta de manera positiva a la generación del modelo As-built cuando se usan las nubes de puntos, ya que los tiempos de creación de los objetos se reducen considerablemente; esto se debe a que es el algoritmo el que realiza el trabajo pesado, donde logra convertir los archivos a formatos útiles para el proceso de entrenamiento y posterior validación. Este proceso tomó entre minuto y treinta segundos (1:30 min) a dos minutos (2:00 min) para generar una

identificación del objeto, indicar el porcentaje de asertividad y al mismo tiempo crear el objeto 3D para descargarlo y usarlo en el software de modelado.

El aplicativo propuesto y su estructuración pueden lograr un nivel alto de precisión, permitiendo que el proceso de generación de objetos 3D a partir de nubes de puntos; logre un alto nivel de automatización y se convierta en una herramienta que apoye a todos los equipos de proyectos.

8. RECOMENDACIONES

Se recomienda que el levantamiento de las nubes de puntos este estandarizado incluyendo resolución, formatos de intercambio de datos, distancia entre los puntos, todo esto para garantizar la homogeneidad de los datos y la calidad de estos, así como también la compatibilidad de los datos con softwares BIM.

Es recomendable que estén definidos los criterios de nivel de información necesario (LOIN) para cada objeto, de esta manera los resultados obtenidos cada vez serán más acertados ya que el algoritmo validará los mismos criterios, reduciendo la incertidumbre.

Así mismo las organizaciones deben capacitar a más personas en el procesamiento y uso de los datos, esto con el fin de usar las nuevas tecnologías de digitalización de activos.

Este campo de investigación es amplio y se encuentra en una etapa temprana de desarrollo y validación, en cierta medida porque todas las industrias son diferentes, por esta razón es recomendable en proyectos futuros emplear más técnicas de inteligencia artificial buscando una solución estandarizada para todas las industrias del sector de infraestructura.

REFERENCIAS

- [1] C. Eastman, J. Lee, Y. Jeong, y J. Lee, «Automatic rule-based checking of building designs», *Autom Constr*, vol. 18, n.º 8, pp. 1011-1033, dic. 2009, doi: 10.1016/j.autcon.2009.07.002.
- [2] M. Shao, M. Barabash, O. Bashynska, Y. Bashynskiy, y A. Bieliatynskiy, «Building constructions calculation models of reinforced concrete using BIM technologies», *Ain Shams Engineering Journal*, vol. 15, n.º 9, 2024, doi: 10.1016/j.asej.2024.102894.
- [3] Q. Nguyen, H.-B. Nguyen, y V. Mai, «Developing BIM Objects Libraries for Provision of BIM Services: An Action Research», *International Journal of Sustainable Construction Engineering and Technology*, vol. 15, n.º 3, pp. 105-117, 2024, doi: 10.30880/ijscet.2024.15.03.010.
- [4] R. Hu *et al.*, «Using Airborne Laser Scanner and Path Length Distribution Model to Quantify Clumping Effect and Estimate Leaf Area Index», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, n.º 6, pp. 3196-3209, jun. 2018, doi: 10.1109/TGRS.2018.2794504.
- [5] C. P. Vavoulidis y I. Pitas, «Morphological iterative closest point algorithm», *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1296, pp. 416-423, 1997, doi: 10.1007/3-540-63460-6_145.
- [6] D. F. Laefer y L. Truong-Hong, «Toward automatic generation of 3D steel structures for building information modelling», *Autom Constr*, vol. 74, pp. 66-77, 2017, doi: 10.1016/j.autcon.2016.11.011.
- [7] N. Decker, Y. Wang, y Q. Huang, «Efficiently registering scan point clouds of 3D printed parts for shape accuracy assessment and modeling», *J Manuf Syst*, vol. 56, pp. 587-597, 2020, doi: 10.1016/j.jmsy.2020.04.001.
- [8] L. Ma, J. Zhu, L. Yang, H. Liu, Y. Fan, y S. Yang, «A point cloud optimization method for accurate line-scan 3-D measurement in motion with dynamic disturbance», *Measurement (Lond)*, vol. 231, 2024, doi: 10.1016/j.measurement.2024.114669.
- [9] H. Luan *et al.*, «Challenges and Future Directions of Big Data and Artificial Intelligence in Education», *Front Psychol*, vol. 11, 2020, doi: 10.3389/fpsyg.2020.580820.
- [10] C. Liu, G.-C. Wang, y H.-F. Wang, «The Application of Artificial Intelligence in Engineering Education: A Systematic Review», *IEEE Access*, vol. 13, pp. 17895-17910, 2025, doi: 10.1109/ACCESS.2025.3532595.
- [11] M. Younas, D. Abdel Salam El-Dakhs, y Y. Jiang, «A Comprehensive Systematic Review of AI-Driven Approaches to Self-Directed Learning», *IEEE Access*, vol. 13, pp. 38387-38403, 2025, doi: 10.1109/ACCESS.2025.3546319.
- [12] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, y B. Poole, «Score-Based Generative Modeling through Stochastic Differential Equations», *ICLR 2021 - 9th International*

Conference on Learning Representations, nov. 2020, Accedido: 18 de abril de 2025. [En línea]. Disponible en: <https://arxiv.org/abs/2011.13456v2>

- [13] J. Du, H. Huang, Q. Tan, Y. Li, L. Ding, y F. Shuang, «An Efficient Point Cloud Correlation Enhancement RCNN for 3D Object Detection», *Information Technology and Control*, vol. 54, n.º 1, pp. 198-218, 2025, doi: 10.5755/j01.itc.54.1.35616.
- [14] T. Huang, R. Gao, H. Wang, J. Liu, y S. Li, «mBox: 3D object detection based on millimeter-wave radar», *Measurement (Lond)*, vol. 246, 2025, doi: 10.1016/j.measurement.2024.116568.
- [15] P. Giannitrapani, T. Cattai, y S. Colonnese, «Point cloud enhancement by projection sampling», *EURASIP J Image Video Process*, vol. 2025, n.º 1, 2025, doi: 10.1186/s13640-025-00665-4.
- [16] «Machine Learning & Clustering: el algoritmo DBSCAN». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://datascientest.com/es/machine-learning-clustering-dbscan>
- [17] Z. Xiang, A. Rashidi, y G. Ou, «Integrating Inverse Photogrammetry and a Deep Learning–Based Point Cloud Segmentation Approach for Automated Generation of BIM Models», *J Constr Eng Manag*, vol. 149, n.º 9, sep. 2023, doi: 10.1061/jcemd4.coeng-13020.
- [18] H. Jiang *et al.*, «Merged-nets enumeration for the systematic design of multicomponent reticular structures», *J BioX Res*, vol. 386, n.º 6722, pp. 659-666, 2024, doi: 10.1126/science.ads7866.
- [19] Y. Liu, H. Huang, G. Gao, Z. Ke, S. Li, y M. Gu, «Dataset and benchmark for as-built BIM reconstruction from real-world point cloud», *Autom Constr*, vol. 173, 2025, doi: 10.1016/j.autcon.2025.106096.
- [20] «Se espera que el mercado de software de modelado de información de construcción (BIM) crezca a 21 mil millones de dólares para 2025 - Cambashi». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://cambashi.com/building-information-modeling-bim-software-market-expected-to-grow-to-21bn-by-2025/>
- [21] H. A. Altuwaijri, A. Al Kafy, y Z. A. Rahaman, «Multi-temporal remote sensing and geospatial analysis for urban ecosystem service dynamics: A three-decade assessment of land surface transformation in Jeddah, Saudi Arabia», *Physics and Chemistry of the Earth, Parts A/B/C*, vol. 139, p. 103892, sep. 2025, doi: 10.1016/J.PCE.2025.103892.
- [22] «Conjunto de herramientas Plant 3D de Autodesk AutoCAD | Características». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://www.autodesk.com/latam/products/autocad/included-toolsets/autocad-plant-3d>
- [23] «Download Undet for Revit to reach better quality | Undet». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://www.undet.com/download-undet-revit/>
- [24] «NeRF 3D: Cómo crear entornos fotorrealistas con IAPlain Concepts». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://www.plainconcepts.com/es/nerf-3d/>
- [25] «CloudCompare Workflow — Dot3D». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://www.dot3d.app/cloudcompare>
- [26] «MeshLab: descarga e instalación gratuitas en Windows | Microsoft Store». Accedido: 20 de octubre de 2025. [En línea]. Disponible en:

<https://apps.microsoft.com/detail/9n4t15dxc6c0?launch=true&mode=full&hl=es-es&gl=co&ocid=bingwebsearch>

- [27] «iTwin Capture Manage & Extract | Bentley Systems | Infrastructure Engineering Software Company». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://www.bentley.com/software/itwin-capture-manage-extract/>
- [28] «EdgeWise | As-Built Scan-to-BIM Software | ClearEdge3D». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://www.clearedge3d.com/products/edgewise/>
- [29] «Trimble RealWorks | Office Software | Trimble Geospatial». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://geospatial.trimble.com/en/products/software/trimble-realworks>
- [30] «Sistema de Transmisión Nacional STN». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://www.xm.com.co/transmisi%C3%B3n/stn-sistema-de-transmision-nacional>
- [31] I. Gibson, T. Kvan, y L. W. Ming, «Rapid prototyping for architectural models», *Rapid Prototyp J*, vol. 8, n.º 2, pp. 91-99, 2002, doi: 10.1108/13552540210420961.
- [32] X. Wang, H. Dong, Y. Wang, W. Zhu, y J. Guo, «A Study of Automated Calibration Problems of Lightweight 3D Digital Models for Power Grid Engineering», *International Journal of High Speed Electronics and Systems*, vol. 34, n.º 04, dic. 2025, doi: 10.1142/S0129156425403122.
- [33] S. H. Zargar, J. Sadeghi, M. Hashemi, N. Motallebi, y A. Andaji Garmaroodi, «Introducing a New Method to Convert a 2D Hand-Drawn Perspective to a 3D Digital Model», *Nexus Netw J*, vol. 22, n.º 2, pp. 291-307, 2020, doi: 10.1007/s00004-019-00457-0.
- [34] F. Fakhari Tehrani, J. Absi, F. Allou, y C. Petit, «Investigation into the impact of the use of 2D/3D digital models on the numerical calculation of the bituminous composites' complex modulus», *Comput Mater Sci*, vol. 79, pp. 377-389, 2013, doi: 10.1016/j.commatsci.2013.05.054.
- [35] G. Caruso, M. Carulli, y M. Bordegoni, «Augmented Reality System for the Visualization and Interaction with 3D Digital Models in a Wide Environment», *Comput Aided Des Appl*, vol. 12, n.º 1, pp. 86-95, 2015, doi: 10.1080/16864360.2014.949579.
- [36] T. Fukuda, T. Tokuhara, y N. Yabuki, «A dynamic physical model based on a 3D digital model for architectural rapid prototyping», *Autom Constr*, vol. 72, pp. 9-17, 2016, doi: 10.1016/j.autcon.2016.07.002.
- [37] S. Filippi, B. Motyl, y C. Bandera, «Comparing parametric solid modelling/reconfiguration, global shape modelling and free-form deformation for the generation of 3D digital models of femurs from X-ray images», *Comput Methods Biomech Biomed Engin*, vol. 12, n.º 1, pp. 101-108, 2009, doi: 10.1080/10255840802178582.
- [38] L. T. Mo, M. Huurman, S. P. Wu, y A. A. A. Molenaar, «2D and 3D meso-scale finite element models for ravelling analysis of porous asphalt concrete», *Finite Elements in Analysis and Design*, vol. 44, n.º 4, pp. 186-196, feb. 2008, doi: 10.1016/J.FINEL.2007.11.012.
- [39] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, y M. Nießner, «ScanNet: Richly-annotated 3D reconstructions of indoor scenes», *Proceedings - 30th IEEE Conference on*

Computer Vision and Pattern Recognition, CVPR 2017, vol. 2017-January, pp. 2432-2443, nov. 2017, doi: 10.1109/CVPR.2017.261.

- [40] W. Meeussen *et al.*, «Autonomous door opening and plugging in with a personal robot», *Proc IEEE Int Conf Robot Autom*, pp. 729-736, 2010, doi: 10.1109/ROBOT.2010.5509556.
- [41] H. S. Park, H. M. Lee, H. Adeli, y I. Lee, «A new approach for health monitoring of structures: Terrestrial laser scanning», *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, n.º 1, pp. 19-30, ene. 2007, doi: 10.1111/J.1467-8667.2006.00466.X.
- [42] H. Niska, J. P. Skön, P. Packalén, T. Tokola, M. Maltamo, y M. Kolehmainen, «Neural networks for the prediction of species-specific plot volumes using airborne laser scanning and aerial photographs», *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, n.º 3 PART 1, pp. 1076-1085, 2010, doi: 10.1109/TGRS.2009.2029864.
- [43] M. A. Fischler y R. C. Bolles, «Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography», *Readings in Computer Vision*, pp. 726-740, ene. 1987, doi: 10.1016/B978-0-08-051581-6.50070-2.
- [44] J. Park y Y. K. Cho, «Point Cloud Information Modeling: Deep Learning–Based Automated Information Modeling Framework for Point Cloud Data», *J Constr Eng Manag*, vol. 148, n.º 2, feb. 2022, doi: 10.1061/(ASCE)CO.1943-7862.0002227.
- [45] H. Chen *et al.*, «Multiobjective Optimization of a 3D Laser Scanning Scheme for Engineering Structures Based on RF-NSGA-II», *J Constr Eng Manag*, vol. 149, n.º 2, feb. 2023, doi: 10.1061/(ASCE)CO.1943-7862.0002411.
- [46] T. Gao, B. Akinci, S. Ergan, y J. Garrett, «An approach to combine progressively captured point clouds for BIM update», *Advanced Engineering Informatics*, vol. 29, n.º 4, pp. 1001-1012, oct. 2015, doi: 10.1016/J.AEI.2015.08.005.
- [47] A. Adán, B. Quintana, S. A. Prieto, y F. Bosché, «Scan-to-BIM for ‘secondary’ building components», *Advanced Engineering Informatics*, vol. 37, pp. 119-138, ago. 2018, doi: 10.1016/J.AEI.2018.05.001.
- [48] «500-kV AC Substation Design Criteria Summary of Industry Practices», *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-88, n.º 6, pp. 854-861, 1969, doi: 10.1109/TPAS.1969.292402.
- [49] A. Ericsson y R. Eriksson, «FUTURE SUBSTATION DESIGNS.», *ASEA Journal (Allmaenna Svenska Elektriska Aktiebolaget)*, vol. 56, n.º 3, pp. 12-15, 1983.
- [50] J. Marks, «Speed substation design with digital imaging», *Electrical World*, vol. 210, n.º 8, pp. 39-41, 1996.
- [51] D. Atanackovic, D. McGillis, y F. D. Galiana, «A new tool for substation design», *IEEE Transactions on Power Systems*, vol. 13, n.º 4, pp. 1500-1506, 1998, doi: 10.1109/59.736297.
- [52] A. Akbari, F. Cuervo Camargo, y L. V. Krivova, «MODERNIZATION OF SUBSTATION DESIGN WITH THE FIRST CATEGORY OF CONSUMERS», *International Journal on Technical and Physical Problems of Engineering*, vol. 16, n.º 58, pp. 86-91, 2024.

- [53] F. C. Chan, «Green approach to electrical transmission substation design and implementation», *HKIE Transactions Hong Kong Institution of Engineers*, vol. 10, n.º 1, pp. 1-7, 2003, doi: 10.1080/1023697X.2003.10667894.
- [54] Z. Cheng *et al.*, «Life cycle carbon dioxide emissions of a typical 500 kV substation: design, production, construction, operation, and recycling», *Electrical Engineering*, 2025, doi: 10.1007/s00202-025-03057-z.
- [55] L. Zhang, X. Yu, A. Aboah, y Y. Adu-Gyamfi, «Three-Dimensional Object Detection and High-Resolution Traffic Parameter Extraction Using Low-Resolution LiDAR Data», *J Transp Eng A Syst*, vol. 151, n.º 3, p. 04025001, ene. 2025, doi: 10.1061/JTEPBS.TEENG-8662.
- [56] L. Yang, Y.-C. Lin, H. Cai, y A. Habib, «From Scans to Parametric BIM: An Enhanced Framework Using Synthetic Data Augmentation and Parametric Modeling for Highway Bridges», *Journal of Computing in Civil Engineering*, vol. 38, n.º 3, p. 04024008, feb. 2024, doi: 10.1061/JCCEE5.CPENG-5640.
- [57] «Scopus - Document details - Integrating Advanced Surveying Technologies and Indigenous Knowledge into 3D Building Information Modeling | Signed in». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://www-scopus-com.ezproxy.eafit.edu.co/record/display.uri?eid=2-s2.0-85196067630&origin=resultslist&sort=plf-f&src=s&sot=b&sdt=b&s=TITLE-ABS-KEY%28%22Advanced+Surveying+Technologies%22%29&sessionSearchId=d8b876d3324d697471db41ec58ef3a41&relpos=1>
- [58] R. B. Rusu y S. Cousins, «3D is here: Point Cloud Library (PCL)», *Proc IEEE Int Conf Robot Autom*, 2011, doi: 10.1109/ICRA.2011.5980567.
- [59] S. Filin y N. Pfeifer, «Segmentation of airborne laser scanning data using a slope adaptive neighborhood», *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 60, n.º 2, pp. 71-80, abr. 2006, doi: 10.1016/J.ISPRSJPRS.2005.10.005.
- [60] D. Hu, V. J. L. Gan, y R. Zhai, «Automated BIM-to-scan point cloud semantic segmentation using a domain adaptation network with hybrid attention and whitening (DawNet)», *Autom Constr*, vol. 164, 2024, doi: 10.1016/j.autcon.2024.105473.
- [61] J. W. Ma, J. Jung, y F. Leite, «Deep Learning–Based Scan-to-BIM Automation and Object Scope Expansion Using a Low-Cost 3D Scan Data», *Journal of Computing in Civil Engineering*, vol. 38, n.º 6, p. 04024040, sep. 2024, doi: 10.1061/JCCEE5.CPENG-5751.
- [62] D. Rebolj, Z. Pučko, N. Č. Babič, M. Bizjak, y D. Mongus, «Point cloud quality requirements for Scan-vs-BIM based automated construction progress monitoring», *Autom Constr*, vol. 84, pp. 323-334, dic. 2017, doi: 10.1016/J.AUTCON.2017.09.021.
- [63] T. Kang y J. Kim, «Feature Template–Based Parametric Swept Geometry Generation from Point Cloud Data: Pipeline Design in Building Systems», *J Pipeline Syst Eng Pract*, vol. 14, n.º 1, p. 04022052, oct. 2022, doi: 10.1061/(ASCE)PS.1949-1204.0000693.
- [64] «Scopus - Document details - Existing Buildings Recognition and BIM Generation Based on Multi-Plane Segmentation and Deep Learning». Accedido: 18 de abril de 2025. [En línea].

Disponible en: <https://www-scopus-com.ezproxy.eafit.edu.co/record/display.uri?eid=2-s2.0-86000521733&origin=resultslist&sort=plf-f&src=s&sot=b&sdt=b&cluster=scosubjabbr%2C%22ENGI%22%2Ct%2Bscosubtype%2C%22ar%22%2Ct&s=TITLE-ABS-KEY%28%22bim+modelling%22%29&sessionSearchId=fc5819f48c3b502ff5fc7abd7d022fcd>

- [65] J. Feng, L. Yang, y J. Xiao, «Subsurface Object 3D Modeling Based on Ground Penetration Radar Using Deep Neural Network», *Journal of Computing in Civil Engineering*, vol. 37, n.º 6, p. 04023030, ago. 2023, doi: 10.1061/JCCEE5.CPENG-5359.
- [66] D. Jung y D. Park, «Accelerated on-Chip Algorithm Based on Semantic Region-Based Partial Difference Detection for LiDAR-Vision Depth Data Transmission Reduction in Lightweight Controller Systems of Autonomous Vehicle», *Proceedings - 2021 IEEE 14th International Symposium on Embedded Multicore/Many-Core Systems-on-Chip, MCSoc 2021*, pp. 16-22, 2021, doi: 10.1109/MCSOC51149.2021.00011.
- [67] G. Jati, M. Molan, F. Barchi, A. Bartolini, G. Mercurio, y A. Acquaviva, «LIDAROC: Realistic LiDAR Cover Contamination Dataset for Enhancing Autonomous Vehicle Perception Reliability», *IEEE Sens Lett*, vol. 8, n.º 9, 2024, doi: 10.1109/LESENS.2024.3434624.
- [68] M. Sahal, A. O. Kurniawan, y R. E. A. Kadir, «Object Detection for Autonomous Vehicle using Single Camera with YOLOv4 and Mapping Algorithm», *2021 4th International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2021*, pp. 144-149, 2021, doi: 10.1109/ISRITI54043.2021.9702764.
- [69] J. Lambert *et al.*, «Performance Analysis of 10 Models of 3D LiDARs for Automated Driving», *IEEE Access*, vol. 8, pp. 131699-131722, 2020, doi: 10.1109/ACCESS.2020.3009680.
- [70] «Organización y digitalización de la información en edificaciones y obras de ingeniería civil, incluyendo el bim (building information modelling). gestión de la información usando el bim. parte 2: fase de entrega de los activos.» Accedido: 17 de octubre de 2025. [En línea]. Disponible en: <https://tienda.icontec.org/gp-organizacion-y-digitalizacion-de-la-informacion-en-edificaciones-y-obras-de-ingenieria-civil-incluyendo-bim-building-information-modelling-gestion-de-la-informacion-usando-bim-parte-2-fase-de-entrega-de-los-activos-ntc-iso19650-2-2021.html>
- [71] «Intercambio de datos en la industria de la construcción y en la gestión de inmuebles mediante ifc (industry foundation classes). parte 1: esquema de datos.» Accedido: 17 de octubre de 2025. [En línea]. Disponible en: <https://tienda.icontec.org/gp-ntc-iso-intercambio-de-datos-en-la-industria-de-la-construccion-a-n-y-en-la-gestion-de-inmuebles-mediante-ifc-industry-foundation-classes-parte-1-esquema-de-datos-ntc-iso16739-1-2024.html>
- [72] S. Zhu, J. Zhou, L. Cheng, X. Fu, Y. Wang, y K. Dai, «Research on a BIM Model Quality Compliance Checking Method Based on a Knowledge Graph», *Journal of Computing in Civil Engineering*, vol. 39, n.º 1, 2025, doi: 10.1061/JCCEE5.CPENG-5950.
- [73] C. R. Qi, H. Su, K. Mo, y L. J. Guibas, «PointNet: Deep learning on point sets for 3D classification and segmentation», *Proceedings - 30th IEEE Conference on Computer Vision and*

Pattern Recognition, CVPR 2017, vol. 2017-January, pp. 77-85, nov. 2017, doi: 10.1109/CVPR.2017.16.

- [74] «GitHub - charlesq34/pointnet2: PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://github.com/charlesq34/pointnet2>
- [75] C. R. Qi, L. Yi, H. Su, y L. J. Guibas, «PointNet++: Deep hierarchical feature learning on point sets in a metric space», *Adv Neural Inf Process Syst*, vol. 2017-December, pp. 5100-5109, 2017, Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://arxiv.org/pdf/1706.02413.pdf>
- [76] «PointNet++: una arquitectura basada en puntos para datos de nubes de puntos 3D | por Nikita Malviya | Medium». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://medium.com/@nikitamalviya/pointnet-a-point-based-architecture-for-3d-point-cloud-data-32c1e7a16b84>
- [77] «cnns.ipynb - Colab». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: https://colab.research.google.com/github/juansensio/blog/blob/master/043_cnn_arquitecturas/arquitecturas.ipynb
- [78] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, y J. M. Solomon, «Dynamic graph Cnn for learning on point clouds», *ACM Trans Graph*, vol. 38, n.º 5, oct. 2019, doi: 10.1145/3326362.
- [79] F. J. Soler, L. M. Jiménez, D. Valiente, L. Payá, y Ó. Reinoso, «Comparative Analysis of Segmentation Techniques for Reticular Structures», *Proceedings of the International Conference on Informatics in Control, Automation and Robotics*, vol. 1, pp. 413-423, ene. 2023, doi: 10.5220/0012177100003543.
- [80] «DeepLab. DeepLab is a family of convolutional... | by Saba Hesarakı | Medium». Accedido: 20 de octubre de 2025. [En línea]. Disponible en: <https://medium.com/@saba99/deeplab-095f387f891f>
- [81] «DeepLab: Una inmersión profunda en el procesamiento visual avanzado - viso.ai». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://viso.ai/deep-learning/deeplab/>
- [82] «Feature Extraction in Machine Learning: A Complete Guide | DataCamp». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://www.datacamp.com/tutorial/feature-extraction-machine-learning>
- [83] «Deep Convolutional Neural Network (DCNN) Explained | Built In». Accedido: 20 de abril de 2025. [En línea]. Disponible en: <https://builtin.com/articles/dcnn>
- [84] F. Pistilli, G. Fracastoro, D. Valsesia, y E. Magli, «Learning Graph-Convolutional Representations for Point Cloud Denoising», *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12365 LNCS, pp. 103-118, 2020, doi: 10.1007/978-3-030-58565-5_7.
- [85] «DeepLearningExamples/PyTorch/Classification/ConvNets/efficientnet at master · NVIDIA/DeepLearningExamples · GitHub». Accedido: 20 de octubre de 2025. [En línea]. Disponible en:

<https://github.com/NVIDIA/DeepLearningExamples/tree/master/PyTorch/Classification/ConvNets/efficientnet>

- [86] S. Qiu *et al.*, «LiDAR-Simulated Multimodal and Self-Supervised Contrastive Digital Twin Approach for Probabilistic Point Cloud Generation of Rail Fasteners», *Journal of Computing in Civil Engineering*, vol. 39, n.º 2, p. 04025001, ene. 2025, doi: 10.1061/JCCEE5.CPENG-6137.
- [87] «Online FlowChart & Diagrams Editor - Mermaid Live Editor». Accedido: 5 de octubre de 2025. [En línea]. Disponible en:
https://mermaid.live/edit#pako:eNp9U81y2jAQfhWNZnojBINtiA-daSAh_0knnR4qMoxiL6CMLXkkmfyRB-ohL9ArL9a1LCDtIT4wu-j72f0sv9JUZUATOsVY7rg2pIfo4kk-Hxjp1KkQt2Rvb2v5JDdQg4p9pJrYkWpSAYEpNU843cN47BGroIVGTIhy8pOZXUP07KSVhny9PzyD6y7IiMpu_cPhuBPfd5Ahs7ziP0ELWYiRccCx8yVMwTJM0VKu_B6owa8JR85_SsEPwljwVNX5JjdaCi5RrWM1yP5BVCvEFgor3fs9MZsDBJq8BJVuCHF-k-Oe-fgiI2oKzXMAEVSsclh7ARO2FEjrkmac-P2yJQmg84XgqkJSfZJF-slz1sk6pD171K13HiNE6dxysYV11m9PzyoDyk0CpBN3T9m_x6MndY2tQ_otk_nYyC36_f_EzljQ67nn6d76gY524qduf6cXQBownW6EMtdlLu7cO5gFxuDOagCrF6_c9IbeciFglx-mnPUR_OUS0e5YsNNNoJitkmQL223gCVeOcL2NUYOpcovnpg5xE1jKLd7raXNo2g9GSc-_dvwbdqmM1Z5fILPEl4EsmKt6Jw--ceDv7FjIO9qicy0ymIhdQYsWoAtet_S1xk6oXUABE5pgmcGMo--ETuQb0koufyIVbJhaVfMFTWY8N9hVJV5dGak-13wHAZmBHir80GgSOQWavNInmoRxvx3FYa_TH0RhEASDFn2mSRB229FB1I3C3kGvHw66by364iyD9qAX9YM47kTBIA7jbvz2F46zWio

ANEXOS

Anexo A. README

Anexo B. Video_paso_a_paso_instalacion_aplicativo

Anexo C. Captured_images

Anexo D. Archivos_nubes_de_puntos

Anexo E. Equipos_revit

Anexo F. Classification_results

Anexo G. Result_resume

Anexo H. Informe_de_clasificacion