



ESCUELA DE INGENIERÍA EN COMPUTADORES

CE-5302: PROYECTO DE DISEÑO INGENIERÍA EN COMPUTADORES

Documento de Diseño

Generación de nubes de puntos a partir de *stacking* sin información de pose de la cámara

PROFESOR

Luis Diego Noguera Mena

AUTOR

José Julián Camacho Hernández
2019201459

ASESOR

Luis Alberto Chavarría Zamora

13 de marzo de 2024

Tabla de contenidos

1	Fecha de la versión y estatus	1
2	Organización	1
3	Autor	1
4	Historial de cambios	1
5	Introducción	1
5.1	Propósito	2
5.2	Alcance	2
5.3	Contexto	2
5.4	Resumen	3
6	Referencias	4
7	Glosario	5
8	Interesados	5
9	Perspectivas de diseño	6
9.1	Contexto	7
9.2	Composición	9
9.3	Lógica	10
9.4	Dependencias	10
9.5	Información	10
9.6	Uso de patrones	10
9.7	Interfaces	10
9.8	Interfaz de usuario	10
9.9	Estructura	11
9.10	Interacción	11
9.11	Dinámica de estados	12
9.12	Algoritmos	12
9.13	Recursos	14
10	Apéndice: Alternativas de diseño	15
10.1	Alternativa 1	15
10.2	Alternativa 2	17
10.3	Selección de alternativa	18
10.4	Evidencia de envío del documento	19

1. Fecha de la versión y estatus

En primer término, en el cuadro 1 se detalla la versión, fecha y estatus del documento.

Versión	Fecha	Estatus
1	06/03/2024	Versión inicial
2	13/03/2024	Versión aprobada

Cuadro 1: Versión, fecha y estatus del documento.

2. Organización

Instituto Tecnológico de Costa Rica, Escuela de Ingeniería en Computadores.

3. Autor

José Julián Camacho Hernández.

4. Historial de cambios

A continuación, se presenta en el cuadro 2 el historial de cambios del documento.

Versión	Fecha	Responsable	Descripción
1	06/03/2024	José Julián Camacho Hernández	Versión Inicial
2	13/03/2024	José Julián Camacho Hernández	Versión Aprobada

Cuadro 2: Historial de cambios del documento.

5. Introducción

El objetivo primordial de este producto es proporcionar una solución para la generación de nubes de puntos mediante un algoritmo de apilamiento, sin requerir información de pose de la cámara. La generación de este tipo de representaciones resulta crucial en diversas aplicaciones, como la visión por computadora, la realidad aumentada y la percepción robótica. Este producto se enfoca en superar los desafíos asociados con la falta de información respecto a la cámara, ofreciendo una herramienta para la creación de nubes de puntos tridimensionales a partir de datos de entrada.

5.1. Propósito

El propósito de este producto es brindar una solución para la generación de nubes de puntos sin depender de la información de pose de la cámara, al implementar un algoritmo de apilamiento de distintos conjuntos de puntos. Esto permitirá a los usuarios trabajar con conjuntos de datos que carecen de datos precisos de posición y orientación de la cámara. La generación de nubes de puntos resultará más accesible y aplicable en una variedad de contextos, facilitando la investigación y el desarrollo en áreas como el modelado tridimensional.

5.2. Alcance

El alcance del producto de *software* se centra en desarrollar un Generador de Nubes de Puntos. El mismo se delimita a la implementación de un algoritmo de apilamiento de nubes de puntos sin utilizar información de la pose de la cámara, al realizar la combinación de conjuntos de puntos de entrada para producir una nube de puntos global de salida. Este enfoque resulta beneficioso porque permite la obtención de representaciones tridimensionales precisas y detalladas de entornos de manera más económica.

El producto forma parte integral del proyecto titulado “Desarrollo de Aplicaciones de Procesamiento de Vídeo e Imágenes Orientadas hacia Vehículos Autónomos bajo condiciones de visión no ideales para la Realidad Costarricense”. Dicho proyecto consiste en el desarrollo de un sistema de drones que, al explorar una zona, captura vídeos y realiza estimaciones de fondo monocular para obtener mapas de profundidad con los cuales genera nubes de puntos individuales por cada imagen. La contribución clave del presente producto radica en la fase final de este proceso, donde las nubes de puntos se combinan para proporcionar un modelo tridimensional consolidado.

Dado lo anterior, en este enfoque no se incluye la generación directa de nubes de puntos a partir de imágenes. Adicionalmente, se descarta cualquier funcionalidad relacionada con la manipulación o interacción directa con la cámara. Se parte del supuesto de que el *hardware* está en condiciones adecuadas, centrándose así en el desarrollo de un método para combinar y procesar conjuntos de nubes de puntos sin depender de datos específicos de la posición u orientación de la cámara en cada captura.

Para dicho método, los objetivos se centran en su precisión para generar nubes de puntos fieles a los vídeos captados por los drones, y la meta principal consiste en la integración fluida con el sistema, para que de esa manera sea utilizado en diversos escenarios.

5.3. Contexto

En la actualidad, la captura y procesamiento de datos tridimensionales desempeñan un papel fundamental en diversas aplicaciones, desde la robótica hasta reconstrucción en 3D. Este producto se desarrolla en el contexto de la generación de mapas tridimensionales a partir de la captura de vídeos mediante vehículos no tripulados, en el cual surge un desafío crítico relacionado con la combinación eficiente de las nubes de puntos individuales. Este entorno implica la utilización de drones como herramienta de exploración y recolección de datos espaciales, donde la generación precisa de mapas tridimensionales es esencial.

El mapeo con drones es una tecnología del área de la teledetección, que tiene como fin crear mapas en 2D y 3D de un área utilizando datos de sensores situados en un dron o un vehículo aéreo no tripulado (VANT). Esta técnica busca generar mapas geoespaciales que contienen coordenadas de ubicación específicas del mundo real para cada punto de datos. Los datos de mapas embebidos permiten realizar mediciones del mundo real para aplicaciones de construcción, agricultura, planificación urbana y estudios topográficos. [1]

El producto se ubica y se utilizará en el ambiente de la fotogrametría, que es una de las técnicas más comunes de mapeos tridimensionales con drones, junto con el LiDAR (*Light Detection and Ranging*). Esta segunda consiste en un proceso de mapeo con mejores resultados independientemente de la visibilidad del entorno, pero utiliza sensores que son costosos y pesados.

Por su lado, la fotogrametría es un método más sencillo y económico debido a la gran disponibilidad de cámaras. Sin embargo, depende en gran medida de cuán visibles son las características de los datos capturados con cámaras. La visibilidad se ve afectada por la altura a la que navegue el dron y aspectos ambientales tales como oscuridad, nubes y niebla. [1]

Por su relación con la fotogrametría, el producto tiene la finalidad de ser utilizado en el campo de la visión por computadora y procesamiento de imágenes, específicamente en la generación de nubes de puntos tridimensionales. Esta tarea resulta fundamental para procesar y apilar conjuntos de coordenadas a fin de crear mapas en 3D de áreas definidas.

5.4. Resumen

El producto por desarrollar recibe un conjunto de nubes de puntos y a partir de él genera una nube global de salida. Dicha salida se obtiene al llevar a cabo las diferentes etapas en el flujo de funcionamiento del Generador de Nubes de Puntos, que se visualizan en la figura 1. En este proceso, el producto debe realizar la lectura de un archivo de configuración que contendrá información de las rutas donde se encuentran las nubes de puntos de entrada y de salida, así como valores para ajustar parámetros del algoritmo.

Una vez la lectura fue exitosa, el producto ejecuta un procesamiento previo de las nubes de puntos de entrada con el fin de eliminar ruido, hacer un submuestreo, normalizar los datos, entre otras técnicas para ajustar las nubes de puntos y así generar mejores resultados.

Seguidamente, el Generador lleva a cabo el algoritmo de apilamiento con las nubes de puntos preprocesadas. Dicho algoritmo consiste en la aplicación de un método de optimización conocido como *Iterative Closest Point* (ICP). En este aplican técnicas para conocer la correspondencia entre puntos en el espacio, y de esa manera, alinear y combinar múltiples nubes de puntos con el objetivo de crear una representación tridimensional unificada del entorno.

Para finalizar el proceso, el producto se encarga de generar un archivo *pcd*, en el cual se almacena la información relacionada con la nube de puntos global. Dicho archivo puede ser utilizado posteriormente para su visualización y análisis.

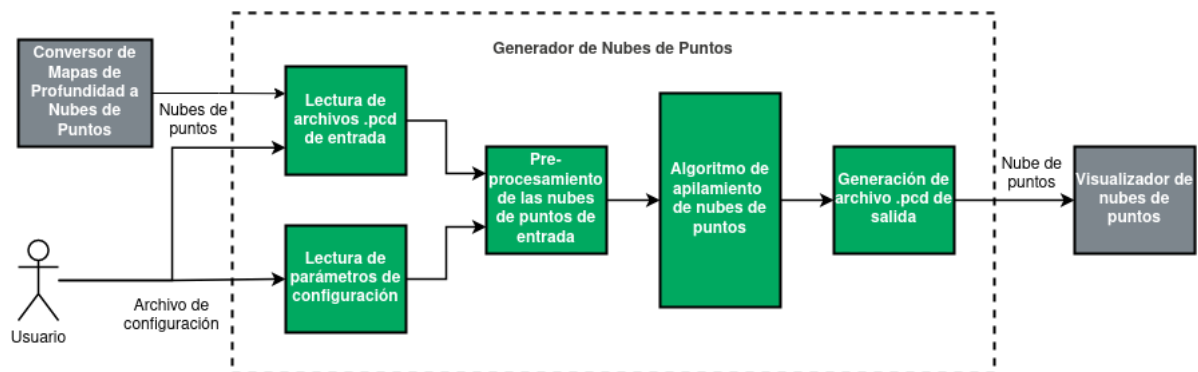


Figura 1: Flujo de funcionamiento del producto de *software*.

Adicionalmente, el producto por desarrollar tiene dos casos de uso principales, los cuales consisten en la aplicación del algoritmo descrito previamente por parte del Sistema de Mapeo 3D con Vehículos Autónomos o del usuario. Un caso de uso secundario es cuando el usuario o la persona supervisora interactúa con el sistema para visualizar los resultados.

El producto cuenta con los módulos necesarios para ejecutar el procedimiento requerido para la generación de la nube de puntos global. Las descripciones, tanto de la estructura como del comportamiento en la interacción de dichos componentes, se detallan en las secciones de perspectivas de diseño.

Como apéndices al documento, se muestran dos alternativas de diseño. La primera consiste en la utilización de redes neuronales para la generación de nubes de puntos, mientras que la segunda presenta opciones de implementar el ICP. Se detalla la justificación para seleccionar una de las opciones.

6. Referencias

- [1] MathWorks. *"What Is Drone Mapping?"* [En línea]. Disponible en: <https://mathworks.com/discovery/drone-mapping.html>. 2024.
- [2] V. Sarode. *"What Are Point Clouds?"* [En línea]. Disponible en: <https://medium.com/analytics-vidhya/what-are-point-clouds-3655d565e142>. 2020.
- [3] L. Brynte. *"Learning and optimizing camera pose"*. [En línea]. Disponible en: <https://research.chalmers.se/en/publication/539208> Accedido el 21 de febrero de 2024. 2024.
- [4] M Abuolaim A y Brown. *"Defocus Deblurring Using Dual-Pixel Data"*. [En línea]. Disponible en: <https://arxiv.org/pdf/2005.00305.pdf> Accedido el 21 de febrero de 2024. 2020.
- [5] Looking Glass. *"Depth Maps: How Software Encodes 3D Space"*. [En línea]. Disponible en: <https://lookingglassfactory.com/blog/depth-map> Accedido el 21 de febrero de 2024. 2023.

- [6] V. Simon. "*PCD: Efficient 3D point cloud data storage and processing*". [En línea]. Disponible en: <https://www.cadinterop.com/en/formats/cloud-point/pcd.html>. Accedido el 21 de febrero de 2024. 2023.
- [7] Maja Boström. *Point Cloud Registration using both Machine Learning and Non-learning Methods: with Data from a Photon-counting LIDAR Sensor*. 2023. URL: <https://liu.diva-portal.org/smash/get/diva2:1761482/FULLTEXT01.pdf>.
- [8] P. J. Besl y N. D. McKay. «A Method for Registration of 3-D Shapes». En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.2 (1992), págs. 239-256. URL: <https://ieeexplore.ieee.org/document/121791>.
- [9] Xiaoshui Huang et al. «A comprehensive survey on point cloud registration». En: *arXiv preprint arXiv:2103.02690* (2021). URL: <https://arxiv.org/pdf/2103.02690.pdf>.

7. Glosario

- Nube de puntos: Es un conjunto desordenado de puntos tridimensionales en un marco de referencia (sistema de coordenadas cartesianas) en la superficie de objetos. [2]
- Pose de cámara: Corresponde a la posición y orientación de una cámara en el espacio tridimensional. [3]
- Estimación de fondo monocular: Es la tarea de estimar el valor de profundidad (distancia relativa a la cámara) de cada píxel dado una única imagen RGB (monocular). [4]
- Mapa de profundidad: Contiene información sobre la distancia de los objetos desde una perspectiva o punto de referencia específico (como una lente de cámara). [5]
- Point Cloud Data: Consiste en un formato de archivo que brinda la capacidad de almacenar y procesar conjuntos de datos organizados de nubes de puntos. [6]
- RANSAC (*Random Sample Consensus*): Es un algoritmo presentado por Fischler y Bolles (1981) que encuentra de manera iterativa una transformación T para registrar una nube de puntos. El algoritmo es no determinista, iterativo y elimina valores atípicos. [7]

8. Interesados

A continuación, se presentan los involucrados y usuarios del producto identificados.

- **Involucrados:**

- Escuela de Ingeniería en Computadores: es la entidad educativa que respalda y supervisa el proyecto. Proporciona el marco institucional para el desarrollo del

proyecto, asegurando que se alinee con los objetivos académicos y los estándares de calidad establecidos por la escuela.

El interés principal de la Escuela de Ingeniería en Computadores radica en la contribución del proyecto al desarrollo académico y la investigación. Además, busca impulsar la formación práctica del estudiante involucrado en el proyecto.

- **Profesor Asesor:** desempeña un papel crucial proporcionando orientación técnica y académica al desarrollador del proyecto. Supervisa y evalúa el progreso, asegurando que el proyecto cumpla con los estándares académicos y de calidad establecidos.

El profesor asesor tiene un interés particular en el éxito académico y técnico del proyecto. Busca fomentar el aprendizaje práctico, la aplicación de conocimientos teóricos y la contribución significativa al campo de estudio relacionado con la estimación de fondo monocular.

- **Desarrollador:** es la persona encargada de la implementación técnica del proyecto, desde el diseño hasta la entrega final del producto.

El interés del desarrollador se centra en la implementación exitosa del proyecto, la aplicación práctica de habilidades de programación y el logro de los objetivos definidos. Adicionalmente, busca aprender y mejorar sus habilidades a través de la experiencia práctica en un entorno académico y de investigación.

- **Área de Investigación Estimación de Fondo Monocular:** este campo de investigación proporciona el contexto y la base teórica para el desarrollo del proyecto. Sus avances y conocimientos influyen en la dirección y enfoque del proyecto, ya que se basa en los principios y descubrimientos del área de estudio.

El interés de esta área radica en la aplicación práctica de los conceptos teóricos y métodos desarrollados en el ámbito de la estimación de fondo monocular. Busca contribuir a la evolución y mejora de las técnicas utilizadas en proyectos con aplicaciones prácticas y significativas.

■ **Usuarios:**

Los usuarios serán, en principio, el profesor asesor Luis Alberto Chavarría Zamora, que es el director del proyecto “Desarrollo de Aplicaciones de Procesamiento de Vídeo e Imágenes Orientadas hacia Vehículos Autónomos bajo condiciones de visión no ideales para la Realidad Costarricense”, del cual es parte el presente producto; así como quien utilice dicho proyecto.

Adicionalmente, se identifica como usuario cualquier persona con interés en generar nubes de puntos globales a partir de nubes individuales por medio del producto, ya sea para fines académicos, en la industria, entre otros.

9. Perspectivas de diseño

En la presente sección se describen las perspectivas o puntos de vista de diseño del producto por desarrollar.

9.1. Contexto

A continuación, se presenta una descripción de los servicios ofrecidos por el producto diseñado. Se detalla la perspectiva contextual del producto mediante el diagrama de casos de uso según el estándar UML, que se visualiza en la figura 2. De esta figura se derivan los siguientes casos de uso:

■ Caso de uso 1:

- **Nombre:** Aplicación del algoritmo de *stacking* por parte del Sistema de Mapeo 3D con Vehículos Autónomos.
- **Actores:**
 - Sistema de Mapeo 3D con Vehículos Autónomos: Corresponde al sistema del cual es parte el presente producto.
 - Sistema de archivos: Corresponde al sistema que gestiona y almacena los archivos.
- **Entradas:** Nubes de puntos en archivos de formato *pcd*.
- **Salidas:** Nube de puntos global en un archivo de formato *pcd*.
- **Precondiciones:** El Conversor de Mapas de Profundidad a Nubes de Puntos, que es la etapa previa en el sistema del cual es parte el presente producto, generó correctamente las nubes de puntos de entrada.
- **Descripción:** Este caso de uso representa la aplicación del algoritmo de *stacking* por parte del Sistema de Mapeo 3D con Vehículos Autónomos. Para llevarlo a cabo, dicho sistema brinda como insumo las puntos individuales generadas en etapas previas y utiliza el presente producto para realizar la combinación de las mismas. Finalmente, se produce una nube de puntos global que representa la escena tridimensional capturada.

■ Caso de uso 2:

- **Nombre:** Aplicación del algoritmo de *stacking* por parte del usuario.
- **Actores:**
 - Usuario: corresponde a la persona usuaria del producto. La persona supervisora, en este caso, tendría la misma interacción de caso de uso que un usuario convencional.
 - Sistema de archivos.
- **Entradas:**
 - Nubes de puntos en archivos de formato *pcd*.
 - Archivo de configuración.
- **Salidas:** Nube de puntos global en un archivo de formato *pcd*.

- **Precondiciones:** Las nubes de puntos de entrada y el archivo de configuración cuentan con el formato correcto.
 - **Descripción:** Este caso de uso implica la aplicación directa del algoritmo de *stacking* por parte del usuario, quien selecciona las nubes de puntos individuales y, opcionalmente, ajusta el proceso mediante un archivo de configuración. La ejecución del algoritmo resulta en la generación de una nube de puntos global que representa la combinación coherente de las nubes de puntos de entrada.
- **Caso de uso 3:**
- **Nombre:** Visualización y evaluación de resultados.
 - **Actores:**
 - Usuario.
 - Persona supervisora: Corresponde al profesor asesor encargado de la evaluación del producto.
 - **Entradas:** Nube de puntos global en un archivo de formato *pcd*.
 - **Salidas:** Visualización y/o evaluación de los resultados.
 - **Precondiciones:** El usuario ha aplicado el algoritmo de *stacking* y ha obtenido la nube de puntos global.
 - **Descripción:** Este caso de uso implica que el usuario visualice los resultados generados por el algoritmo de *stacking*. Si el usuario es la persona supervisora, se encarga de visualizar el resultado y realizar una evaluación del mismo.

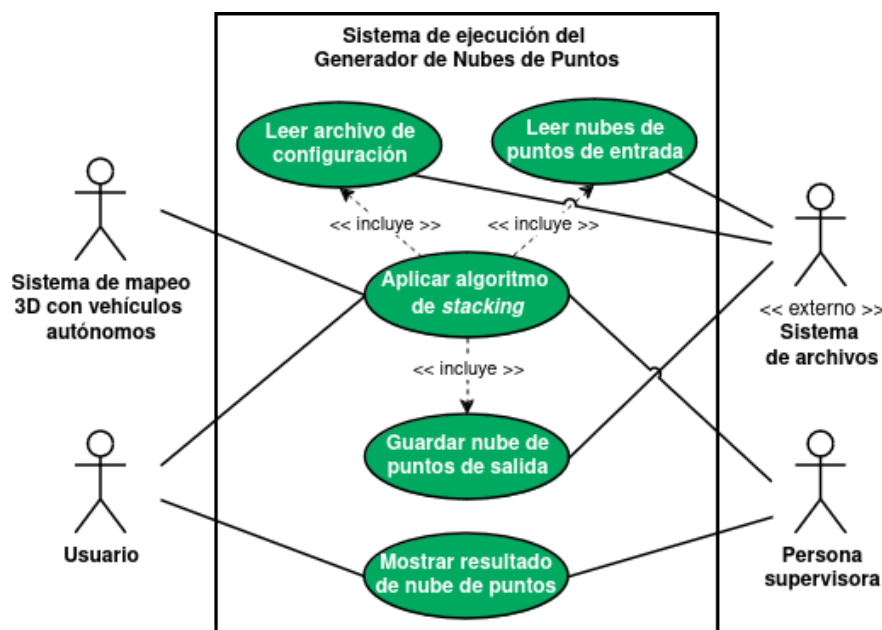


Figura 2: Diagrama de casos de uso UML del producto de *software* por desarrollar.

9.2. Composición

Seguidamente, se detalla la perspectiva de composición del producto. Esta se documenta mediante el diagrama de componentes UML, que se presenta en la figura 3. A continuación, se describen cada uno de los componentes del sistema:

- **Lector de configuraciones:** Este componente se encarga de leer y procesar las configuraciones necesarias para la ejecución del algoritmo de *stacking*. Puede incluir parámetros como umbrales de aceptación, rutas de entrada y de salida, entre otros parámetros de ajuste.
- **Lector de nubes de puntos:** Este componente se encarga de leer las nubes de puntos individuales provenientes de archivos en formato *pcd*. Su función principal es cargar los datos para que sean procesados por el resto del sistema. Ambos componentes lectores cuentan con interfaces tanto hacia el exterior como con el preprocesador.
- **Preprocesador de nubes de puntos:** El preprocesador se encarga de realizar operaciones iniciales sobre las nubes de puntos, como filtrado de ruido, submuestreo, normalización, entre otras técnicas de mejora de la calidad de los datos.
- **Procesador de algoritmo de *stacking*:** Se encarga de apilar los conjuntos preprocesados.
 - **Búsqueda de correspondencias:** Realiza la identificación de puntos correspondientes entre las nubes de puntos individuales, estableciendo relaciones clave para la posterior alineación.
 - **Alineación de nubes de puntos:** Utiliza técnicas como el *ICP* para alinear las nubes de puntos identificadas en la etapa de búsqueda de correspondencias, logrando una combinación coherente y precisa.
- **Generador de archivo de salida:** Este componente se encarga de generar un archivo de salida en formato *pcd* que contiene la nube de puntos global combinada. Dicha nube proviene de su interfaz con el Procesador del algoritmo. Proporciona un formato accesible y ampliamente utilizado para su posterior análisis o visualización.
- **Visualizador de resultados:** Ofrece una interfaz gráfica o herramienta para visualizar los resultados del proceso de *stacking*. Su interfaz con el Generador es el archivo *pcd* de salida. Permite a los usuarios evaluar la calidad y coherencia de la nube de puntos global generada.

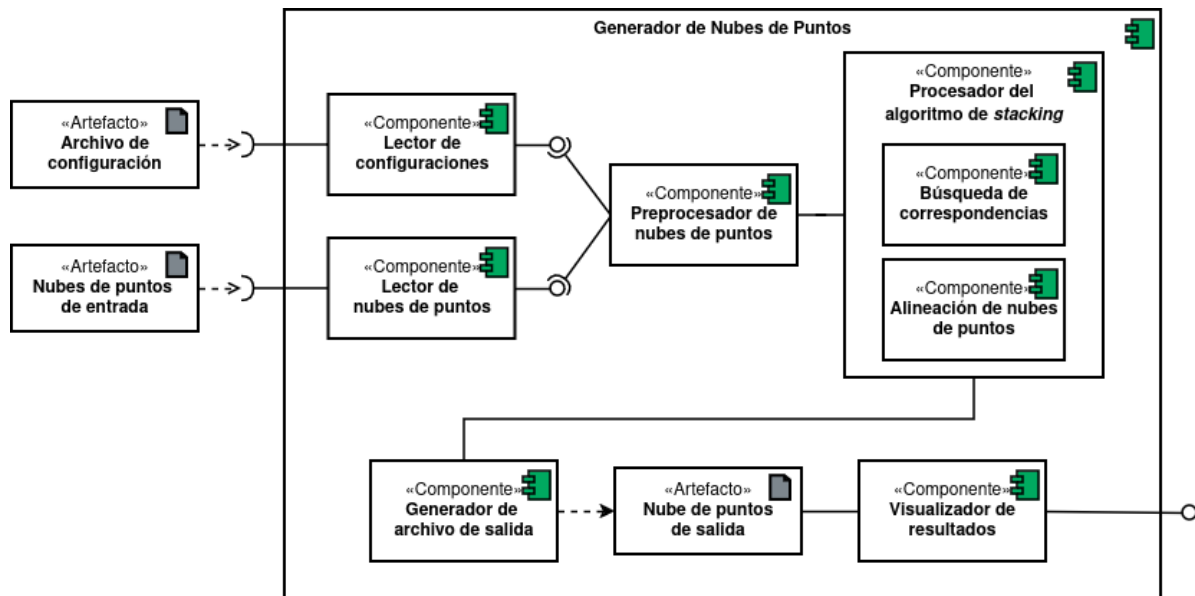


Figura 3: Diagrama de componentes UML del producto a desarrollar.

9.3. Lógica

No aplica.

9.4. Dependencias

No documentado.

9.5. Información

No aplica.

9.6. Uso de patrones

No aplica.

9.7. Interfaces

No documentado.

9.8. Interfaz de usuario

La principal interfaz del producto con el usuario será la terminal, en la cual se ejecutará el programa de la solución. A continuación, se listan las entradas válidas a través de dicha interfaz, así como la salida resultante del algoritmo. Adicionalmente, estas se observan en la figura 3.

- Entradas válidas:
 - Archivos en formato *pcd* que contienen nubes de puntos individuales.
 - Archivo de configuración con parámetros específicos del algoritmo de *stacking*.
- Salidas resultantes:
 - Archivo en formato *pcd* que representa la nube de puntos global generada después del proceso de *stacking*.

9.9. Estructura

No aplica.

9.10. Interacción

A continuación, en la figura 4, se presenta el diagrama de secuencia UML del producto por desarrollar. En este, en color verde se representan las entidades del sistema, y se visualiza el actor principal, que puede ser una persona usuaria o el Sistema de Mapeo 3D con Vehículos Autónomos. Asimismo, se detallan las interacciones entre dichos elementos.

La interacción inicial es la del usuario con el producto por medio del Lector de Archivos de Entrada. Esta se presenta cuando la persona usuaria brinda la indicación de ejecución del programa y proporciona los archivos de entrada, tanto de configuración como de nubes puntos. Al procesarlos, el Lector determinará si el formato es correcto o no, y mostrará un mensaje según corresponda, finalizando así su interacción con el usuario.

Seguidamente, el Lector inicia su interacción con el Preprocesador, al enviarle las nubes de puntos. Esta entidad realiza operaciones que ajustan los datos de entrada para ser emitidos hacia el Procesador del Algoritmo de *Stacking*.

El Procesador se encarga de aplicar iterativamente técnicas de visión por computadora y fotogrametría para buscar correspondencias entre los puntos, y así calcular las rotaciones y traslaciones necesarias para alinear las nubes. En este proceso, si el algoritmo considera que la cantidad de correspondencias entre los conjuntos de datos no es suficiente para ser apilados, comunicará al usuario dicha situación por medio de un mensaje de error.

En caso de que el algoritmo sea aplicado con éxito, se presentan tres nuevas interacciones. Primeramente, se envía la información de la nube de puntos global desde el Procesador hacia el Generador de Archivo de Salida. En segundo lugar, dicho Generador interactúa con el usuario para desplegarle un mensaje de éxito, que indica la obtención de un archivo en formato *pcd* de salida. Este es emitido en la tercera interacción hacia la entidad Visualizador.

La interacción entre el producto y el usuario finaliza cuando el Visualizador muestra la nube de puntos resultante.

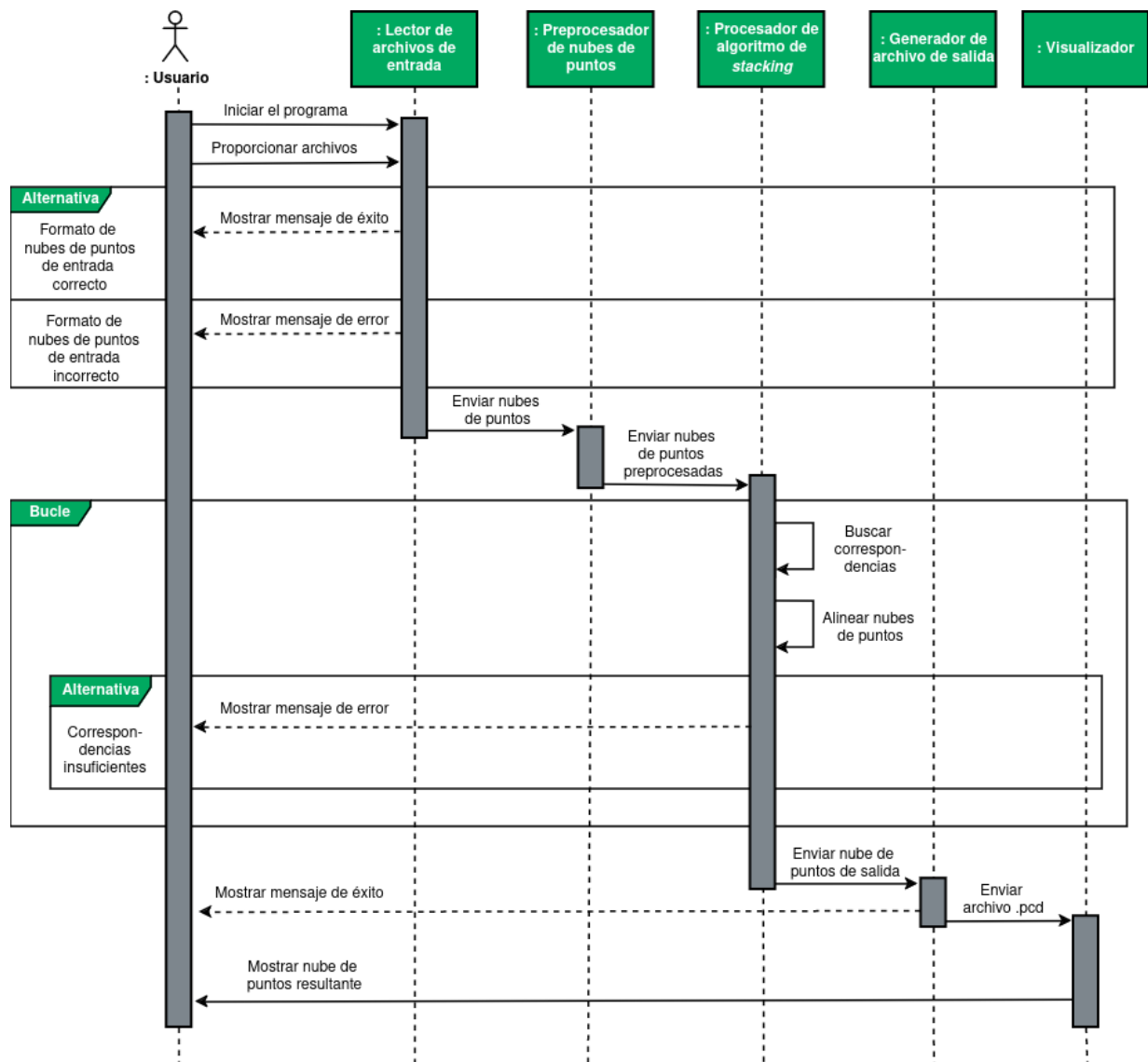


Figura 4: Diagrama de secuencia UML del sistema a desarrollar.

9.11. Dinámica de estados

No aplica.

9.12. Algoritmos

En la presente sección, se describen las operaciones del producto por medio del diagrama de flujo que se visualiza en la figura 5. El algoritmo del producto por desarrollar inicia cuando el usuario da la indicación de ejecutar la solución, y como se presenta en el diagrama, el archivo de configuración es la primera información que se procesa. Este contiene valores que ajustan el algoritmo de *stacking* que se realizará, así como rutas de los archivos de entrada

y salida.

Por lo tanto, el primer método del producto es el que se encarga de leer y procesar el archivo de configuración. Para eso, debe buscarlo en una ruta definida y analizar su contenido. En caso de que dicho archivo cuente con un formato correcto, se utilizarán los valores establecidos y se mostrará al usuario un mensaje de éxito en la lectura. Por el contrario, el programa seleccionará configuraciones por defecto.

El siguiente método es el de la lectura de las nubes de puntos de entrada. Este busca las mismas en las rutas establecidas en el archivo de configuración y las procesa para tomarlas como insumo para las siguientes etapas. Al realizar su lectura se presentan dos casos: si el formato es incorrecto, el programa debe mostrar un mensaje de error y finalizar su ejecución; en caso contrario, se continúa con el flujo de funcionamiento.

En la siguiente etapa, el producto ejecuta un método de preprocesamiento de las nubes de puntos de entrada que fueron leídas. Esta función incluye pasos para la eliminación de ruido, submuestreo, normalización de datos, entre otras técnicas para ajustar las nubes de puntos y así generar mejores resultados al realizar su procesamiento.

Seguidamente, el Generador lleva a cabo el algoritmo principal del producto: el apilamiento de las nubes de puntos preprocesadas. Dicho algoritmo corresponde al *Iterative Closest Point* (ICP). Este es uno de los métodos clásicos para el registro rígido, que es el problema de alinear un conjunto de puntos fuente con un conjunto de puntos objetivo. [8] Este algoritmo alterna entre consultas de puntos más cercanos en el conjunto objetivo y la minimización de la distancia entre puntos correspondientes, garantizando converger a un alineamiento óptimo local.

El flujo general del ICP se presenta en el recuadro punteado en el diagrama de la figura 5. En este, el primer paso es buscar las correspondencias entre puntos cercanos. Seguidamente se pasa al proceso de alinear las nubes. Para esto, se calcula el centro de masa de los puntos y se calculan las rotaciones necesarias para encontrar un resultado. En caso de que no se detecten suficientes correspondencias, el programa debe mostrar un mensaje de error y terminar su ejecución. De lo contrario, se debe revisar la convergencia. Si no se presenta aún, se repiten los pasos descritos previamente, mientras que si existe la convergencia, el programa continúa a los métodos finales.

Para finalizar el proceso, el producto cuenta con un método que se encarga de generar un archivo *pcd*, en el cual se almacena la información relacionada con la nube de puntos global de salida y despliega un mensaje de éxito en el proceso. Asimismo, el programa provee una función para mostrar dicha nube resultante del algoritmo, con el fin de que el usuario realice su visualización y análisis.

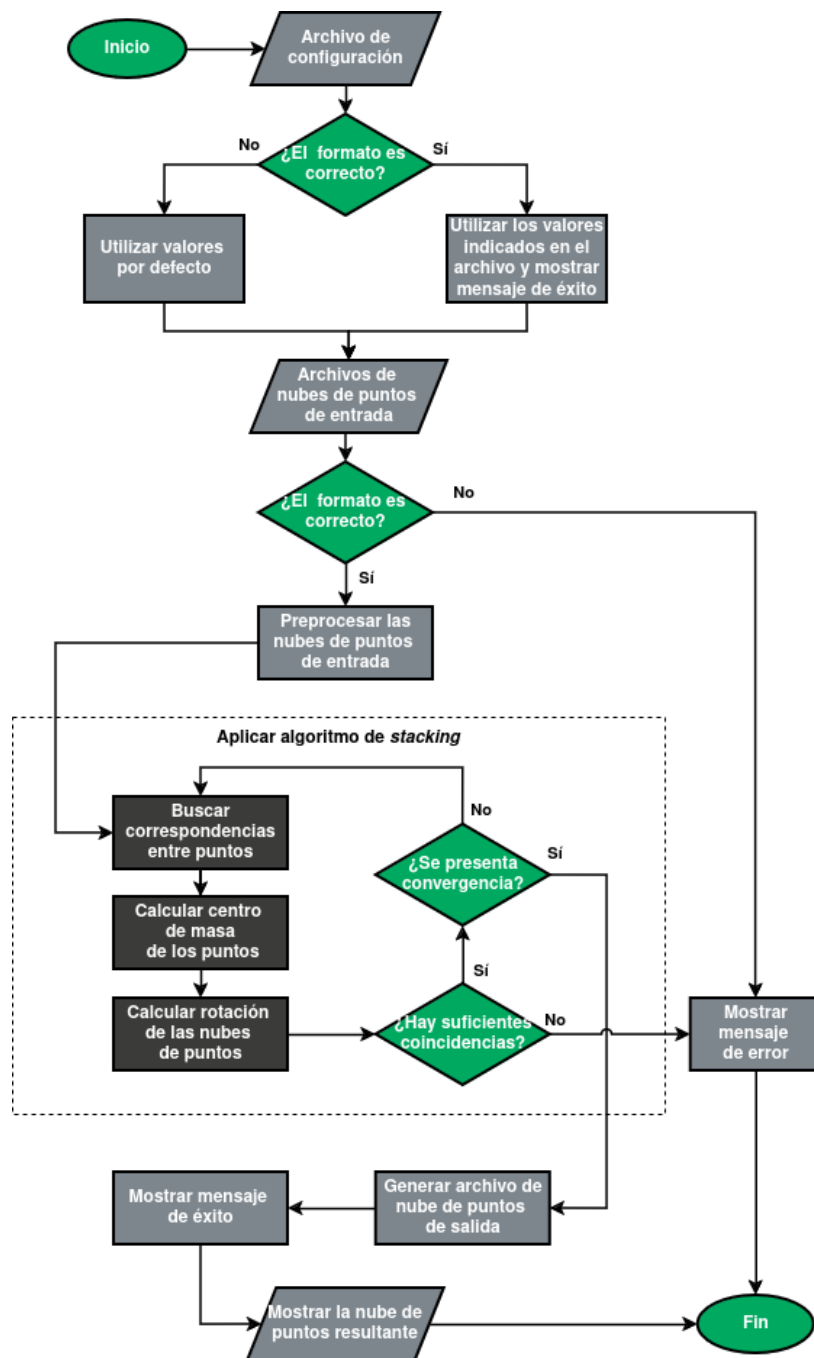


Figura 5: Diagrama de flujo de la perspectiva de algoritmo del producto por desarrollar.

9.13. Recursos

A continuación, por medio del diagrama UML conceptual de clases que se presenta en la figura 6, se detalla la perspectiva de recursos que comprende los siguientes:

- Bibliotecas de manejo matemático: Biblioteca como *Numpy* son fundamentales para el

manejo de operaciones matemáticas y numéricas en *Python*. Proporciona estructuras de datos eficientes, como arreglos y matrices, junto con funciones optimizadas para realizar operaciones algebraicas y estadísticas.

- Bibliotecas de visualización y procesamiento 3D: *Open3D* es una biblioteca dedicada al procesamiento y visualización de datos tridimensionales en entornos de *Python*. Ofrece herramientas para la manipulación de nubes de puntos, reconstrucción 3D, registro y visualización.
- Generador de nubes de puntos a partir de mapas de profundidad: Este componente es responsable de generar conjuntos de puntos utilizando técnicas de procesamiento de imágenes y fotogrametría en el Sistema de Mapeo 3D con Vehículos Autónomos. Su función es crear representaciones tridimensionales que luego serán utilizadas en el proceso de *stacking* para formar una nube de puntos global coherente.
- Sistema de archivos del sistema operativo: Este recurso se encarga de leer y escribir archivos necesarios para el funcionamiento del sistema. Proporciona métodos para la manipulación de archivos, incluyendo la lectura de nubes de puntos individuales y la escritura del resultado final del proceso de *stacking*.

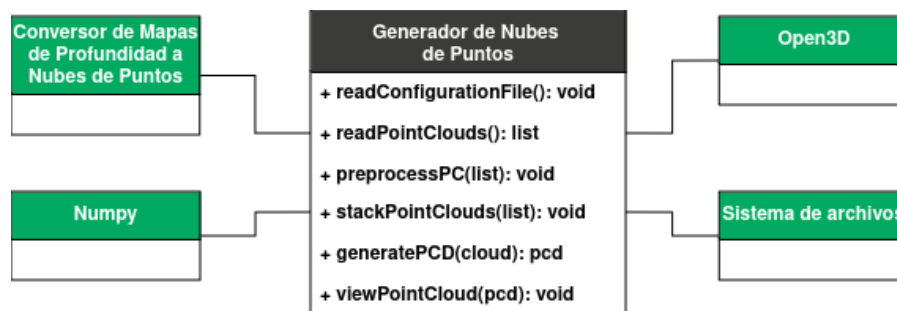


Figura 6: Diagrama conceptual de clases UML del producto por desarrollar.

10. Apéndice: Alternativas de diseño

En el presente apéndice se describe una validación de las alternativas de diseño consideradas para los aspectos de diseño más críticos de la solución.

10.1. Alternativa 1

Utilización de Redes Neuronales para Combinar Nubes de Puntos

En esta alternativa, se propone la utilización de redes neuronales para la generación de nubes de puntos sin información de pose de la cámara, que es el aspecto más crítico

de la solución. La presente propuesta busca aprovechar las capacidades de aprendizaje profundo para capturar patrones complejos y relaciones espaciales entre las nubes de puntos individuales.

La presente solución se basa en los métodos de aprendizaje de características. La estructura de este tipo de *frameworks* se visualiza en la figura 7. Estos, a diferencia de los procedimientos clásicos basados en optimización, utilizan redes neuronales profundas para aprender una búsqueda robusta de correspondencias de características principales en cada nube de puntos de entrada. Luego, la matriz de transformación, necesaria para alinear las nubes de puntos, se finaliza mediante una estimación como lo puede ser el algoritmo RANSAC. [9] De esta manera se abordaría el desafío de la falta de información explícita de pose, ya que el cálculo de dicha matriz brindaría suficiente conocimiento para combinar las nubes de puntos correctamente.

Para llevar a cabo este enfoque, se requiere un conjunto de datos de entrenamiento que contenga ejemplos de pares de nubes de puntos individuales y sus versiones combinadas. Durante el proceso de entrenamiento, cada red neuronal tiene que aprender automáticamente a discernir cómo deben combinarse las nubes de puntos para lograr una representación global coherente. Las entradas para los modelos neuronales serían estos pares de nubes de puntos, mientras que la salida sería la nube de puntos combinada.

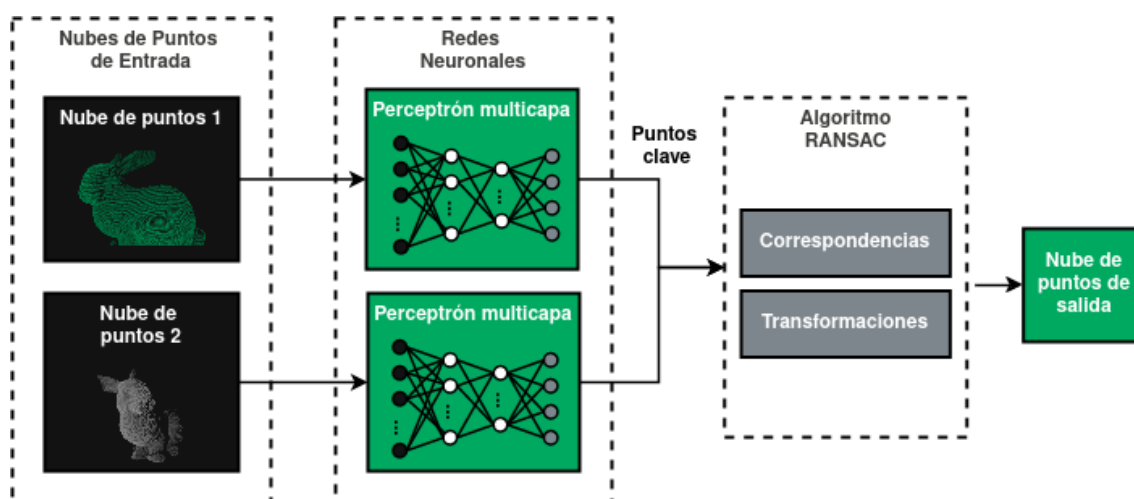


Figura 7: Flujo de funcionamiento del Generador de Nubes de Puntos utilizando redes neuronales.

Para esta alternativa de solución, se propone un modelo de red neuronal profunda (DNN). Esta consiste en un conjunto de capas de entrada y salida con al menos dos capas ocultas. Estas capas ocultas aprenden a realizar transformaciones no lineales de la entrada de la red y se utilizan para el aprendizaje de características. En general, las primeras capas de un extractor de características detectan elementos como bordes, esquinas y colores. [7] En concreto, se presenta la solución con un Perceptrón Multicapa (MLP), que es un tipo específico de DNN que tiene una estructura en capas completamente conectadas. Dicha arquitectura general se presenta en la figura 8 a continuación.

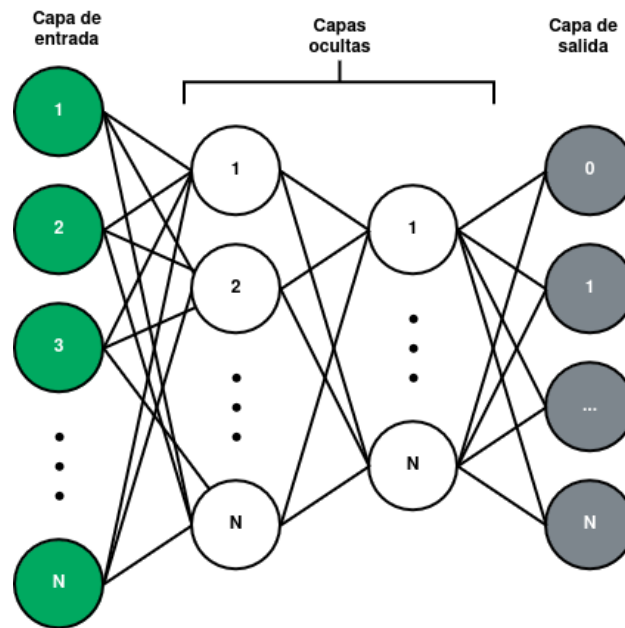


Figura 8: Estructura general de un perceptrón multicapa.

Según investigaciones relacionadas con este tipo de métodos, sus ventajas son las siguientes: [9]

- Los modelos aprendizaje profundo podrían proporcionar una búsqueda de correspondencias robusta y precisa.
- Las correspondencias precisas podrían llevar a resultados precisos de registro mediante el uso de un método simple como RANSAC.

Por otro lado, sus limitaciones se listan a continuación: [9]

- Requieren grandes conjuntos de datos de entrenamiento.
- El rendimiento disminuye drásticamente en escenas desconocidas si estas tienen una gran diferencia de distribución respecto a los datos de entrenamiento.
- Utilizan un proceso de entrenamiento separado para aprender una red independiente.

Esta alternativa de solución se implementaría utilizando bibliotecas de aprendizaje profundo como *TensorFlow* o *PyTorch*. La efectividad de la red neuronal dependerá en gran medida de la calidad y diversidad del conjunto de datos de entrenamiento, lo que impactará directamente en la capacidad de generalización y la calidad de la combinación de nubes de puntos.

10.2. Alternativa 2

Generación de Nubes de Puntos a través del algoritmo ICP

En esta segunda alternativa de diseño, la propuesta se centra en la utilización de un algoritmo basado en optimización, para combinar nubes de puntos sin pose de cámara conocida, específicamente el *Iterative Closest Point* (ICP). La implementación se llevaría a cabo empleando las bibliotecas *Open3D* y *NumPy*, que ofrecen funcionalidades robustas para el procesamiento tridimensional y operaciones matriciales eficientes.

El método que se propone consiste en utilizar estrategias de optimización para estimar la matriz de transformación. La mayoría de estos procedimientos contienen dos etapas: búsqueda de correspondencias y estimación de transformación. La figura 9 resume el proceso principal de esta categoría. [9]

La búsqueda de correspondencias consiste en encontrar el punto coincidente para cada punto en otras nubes de puntos. El siguiente paso consiste en estimar la matriz de transformación utilizando las correspondencias. Estas dos etapas se llevan a cabo de manera iterativa para encontrar la transformación óptima.

Durante el proceso iterativo, las correspondencias pueden no ser precisas al principio, pero se vuelven más precisas a medida que continúa el algoritmo. Luego, la matriz de transformación estimada se vuelve precisa utilizando correspondencias, que será la manera de esta solución de superar la carencia de información de pose de la cámara.

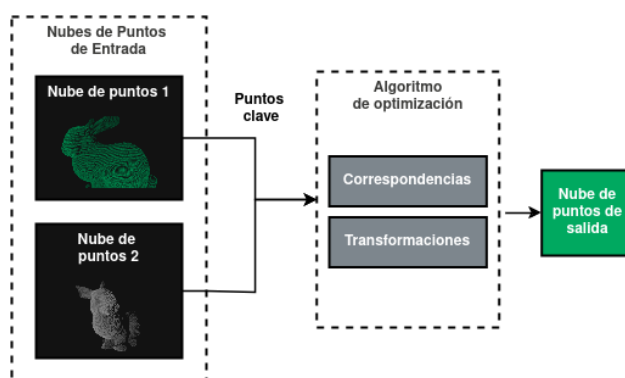


Figura 9: Flujo de funcionamiento de un modelo basado en optimización.

Las ventajas de esta categoría son principalmente dos: [9]

- Existen rigurosas teorías matemáticas pueden garantizar su convergencia.
- No requieren datos de entrenamiento y generalizan bien a escenas desconocidas.

Las limitaciones de esta categoría son que se requieren muchas estrategias sofisticadas para superar las variaciones de ruido, valores atípicos, variaciones de densidad y superposición parcial. [9]

10.3. Selección de alternativa

Para seleccionar la alternativa por desarrollar, se evalúan aspectos técnicos de cada una de las opciones:

Ambas alternativas tienen como objetivo principal la combinación de nubes de puntos sin información de pose de cámara, y son opciones que, a pesar de ser distintas, fueron planteadas de tal manera que fueran viables para cumplir con todos los requerimientos establecidos.

Por otro lado, la implementación de ambas alternativas podría tener impactos positivos en términos de costo total de vida, ya que la implementación exitosa del algoritmo podría reducir los costos asociados con la generación de mapas tridimensionales, especialmente en aplicaciones de drones respecto a las técnicas con LiDAR.

Sin embargo, a grandes escalas del producto, el desarrollo e implementación de múltiples redes neuronales puede requerir recursos físicos adicionales, como *hardware* especializado, *software* propietario y recursos humanos como expertos en aprendizaje profundo para su mantenimiento, lo que podría aumentar significativamente los costos totales de vida para quienes necesitan activamente de las funcionalidades proporcionadas por el producto.

Adicionalmente, la ejecución de diferentes redes neuronales puede requerir más recursos computacionales y energía para entrenar y ejecutar, lo que podría resultar en un mayor consumo de potencia, que a gran escala puede significar un impacto al ambiente. Asimismo, mientras que la solución de ICP puede ejecutarse en *hardware* estándar, la implementación de redes neuronales puede requerir unidades de procesamiento gráfico (GPU) de alta gama o unidades de procesamiento tensorial (TPU), que pueden volverse obsoletas rápidamente y generar desechos tecnológicos cuando se descarten o actualicen, afectando así al medio ambiente si no son tratados correctamente.

Dado que, al comparar ambas soluciones, la alternativa de utilizar el algoritmo basado en optimización de ICP se considera más favorable en términos de conservación al medio ambiente, costo de vida y recursos, y aún cumple con los requerimientos del proyecto, se selecciona esta alternativa como la más adecuada para avanzar con el desarrollo del producto.

10.4. Evidencia de envío del documento

A continuación, en las figuras 10 y 11, se muestra la evidencia de envío del presente documento hacia la persona supervisora.

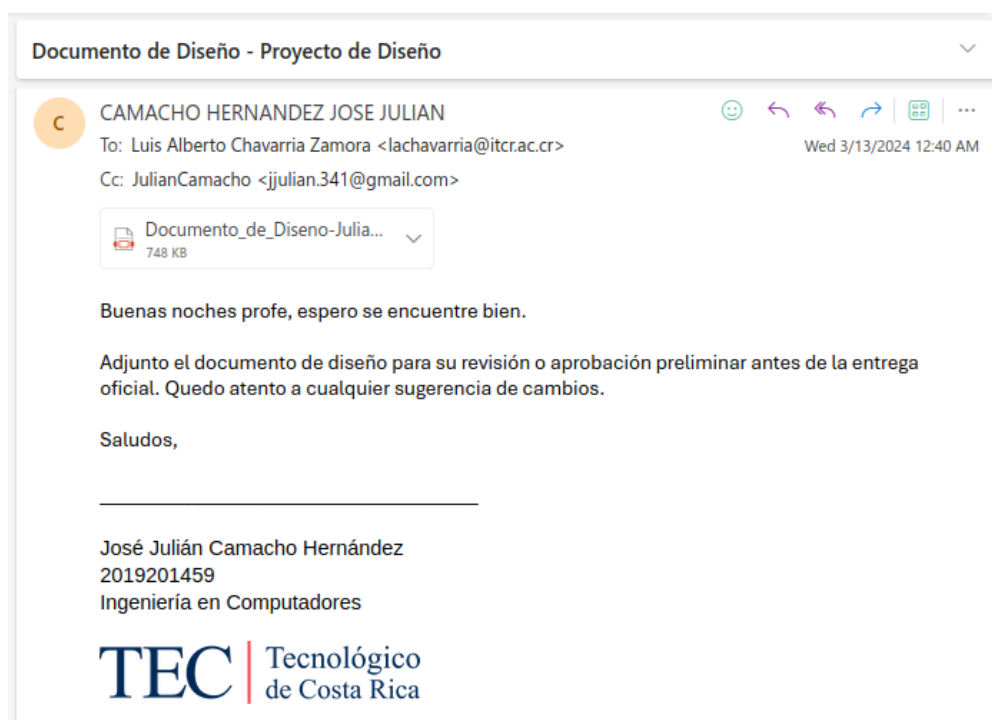


Figura 10: Evidencia de envío al asesor.



Figura 11: Evidencia de recibido por parte del asesor.