



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

Plan de Pruebas

Versión: 0101

Fecha: Mayo/2024

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Computadores
Proyecto de Diseño de Ingeniería en Computadores
I Semestre, 2024
Cartago, Costa Rica

| | | |
|---|--|---------------------------------------|
| TEC Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|---|--|---------------------------------------|

HOJA DE CONTROL

| | | | |
|------------------------|---|----------------------------|------------|
| Organismo | Instituto Tecnológico de Costa Rica | | |
| Proyecto | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | | |
| Entregable | Plan de pruebas | | |
| Autor | José Julián Camacho Hernández | | |
| Versión/Edición | 0101 | Fecha Versión | 29/05/2024 |
| Aprobado por | Luis Alberto Chavarría Zamora | Fecha Aprobación | 29/05/2024 |
| | | Nº Total de Páginas | 21 |

REGISTRO DE CAMBIOS

| Versión doc | Causa del Cambio | Responsable del Cambio | Fecha del Cambio |
|--------------------|-------------------------|-------------------------------|-------------------------|
| 0100 | Versión inicial | José Julián Camacho Hernández | 12/05/2024 |
| 0101 | Versión final | José Julián Camacho Hernández | 29/05/2024 |
| | | | |

ÍNDICE

| | |
|---|-----------|
| <u>1 INTRODUCCIÓN.....</u> | <u>4</u> |
| <u>1.1 Objetivo.....</u> | <u>4</u> |
| <u>1.2 Alcance.....</u> | <u>4</u> |
| <u>2 TRAZABILIDAD DE CASOS DE PRUEBAS – REQUISITOS.....</u> | <u>5</u> |
| <u>3 DEFINICIÓN DE LOS CASOS DE PRUEBAS.....</u> | <u>6</u> |
| <u>4 REPORTE DE RESULTADOS.....</u> | <u>21</u> |

1 INTRODUCCIÓN

1.1 Objetivo

El objetivo de este documento es proporcionar una especificación detallada de los casos de prueba que se utilizarán para verificar y validar el correcto funcionamiento del sistema de *software* desarrollado. Estos casos de prueba están diseñados para cubrir los requisitos funcionales, de usabilidad e interfaces externas identificados en el sistema, asegurando su calidad y fiabilidad antes de su implementación y entrega.

1.2 Alcance

Este documento cubre los casos de prueba que abordan los requisitos funcionales, de usabilidad e interfaces externas del sistema de *software*. Se centra en verificar el comportamiento del sistema en diferentes escenarios y condiciones, así como en garantizar que cumpla con las expectativas de los usuarios finales. Los casos de prueba están diseñados para ser ejecutados durante las etapas de pruebas y aceptación del sistema, con el fin de identificar y corregir posibles errores o fallos antes de su lanzamiento final.

2 TRAZABILIDAD DE CASOS DE PRUEBAS - REQUISITOS

En este apartado se presenta la matriz de trazabilidad de los casos de prueba. Esta indica la correspondencia entre los casos de pruebas definidos, y los requisitos recolectados. Una X la casilla correspondiente, indica que el caso de prueba se encarga de verificar el requisito.

| | CP-1 | CP-2 | CP-3 | CP-4 | CP-5 | CP-6 | CP-7 | CP-8 | CP-9 | CP-10 | CP-11 |
|--------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| RF-1 | | | | | | | | | | | X |
| RF-2 | X | | | | | | | | | | |
| RF-3 | | | X | | | | | | | | |
| RF-4 | | X | | | | | | | | | |
| RF-5 | | | X | | | | | | | | |
| RF-6 | | | | X | | | | | | | |
| RF-7 | | | | | X | | | | | | |
| RF-8 | | | | | | X | | | | | |
| RF-9 | | | | | | | X | | | | |
| RF-10 | | | | | | | | X | | | |
| RU-1 | | | | | | | | | X | | |
| RU-2 | | | | | | | | | X | | |
| RI-1 | | | X | | X | | | | | | |
| RI-2 | X | | X | | | | | | | | |
| RI-3 | | | | | | | | X | | | |
| RD-1 | | | | | X | | | | | | |
| RS-1 | | | | | | | | | | X | |

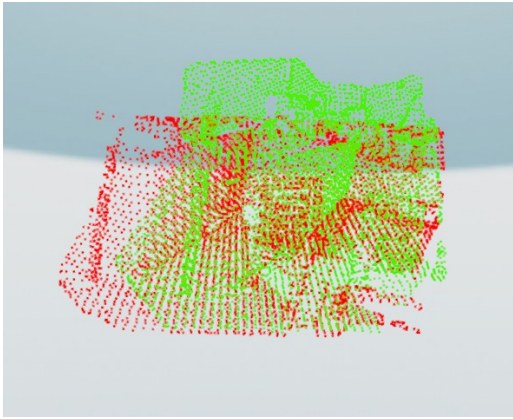
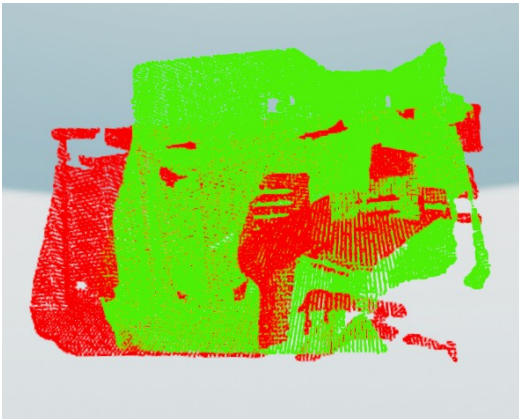
3 DEFINICIÓN DE LOS CASOS DE PRUEBAS

En este apartado se describen en detalle cada uno de los casos de prueba necesarios para verificar la funcionalidad completa del sistema.

| Caso de prueba 1 | Id. Tipo | CP-1 Unitaria |
|---|-------------|------------------|
| Descripción: Verificar que el programa lea correctamente el archivo de configuración y obtenga las rutas de entrada y salida de las nubes de puntos, así como las configuraciones del algoritmo. | | |
| Prerrequisitos Archivo de configuración válido con las rutas de entrada y salida especificadas. <pre data-bbox="560 927 1129 1211"> data > {} config.json > ... 1 { 2 "input_path": "../data/test copy", 3 "config_params": { 4 "voxel_size": 0.02, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.3 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 }</pre> | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa lea correctamente el archivo de configuración. 3. Verificar que el programa obtenga las rutas de entrada y salida de las nubes de puntos, así como las configuraciones del algoritmo. | | |
| Resultado esperado: El programa muestra un mensaje de éxito indicando que se han obtenido las configuraciones del archivo de configuración. | | |
| Resultado obtenido: Al ejecutar el programa, este muestra un mensaje de éxito indicando que se han obtenido las configuraciones del archivo de configuración. <pre data-bbox="229 1733 1436 1912"> • (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src\$ python3 main.py Éxito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/test copy'. Éxito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube_pcd_salida.pcd'. Se cargaron 4 nubes de puntos. WARNING: Using soft CircularBuffer (6144 KiB) FEngine (64 bits) created at 0x617a3c96c190 (threading is enabled) FEngine resolved backend: OpenGL Los parámetros de configuración son: {'voxel_size': 0.02, 'remove_outliers_params': {'nb_neighbors': 20, 'std_ratio': 2.0}, 'combinability_threshold': 0.3}</pre> | | |

| | | |
|--|--|-----------------------------------|
|  Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 2 | Id. | CP-2 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa utilice como ruta de la nube de puntos de salida la indicada en la configuración. | | |
| Prerrequisitos Archivo de configuración válido con la ruta de salida especificada para la nube de puntos. <div data-bbox="555 667 1093 936" data-label="Code-Block"> <pre> data > {} config.json > ... 1 { 2 "input_path": "../data/test copy", 3 "config_params": { 4 "voxel_size": 0.02, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.3 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 }</pre> </div> | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa lea correctamente el archivo de configuración. 3. Ejecutar el proceso de apilamiento de nubes de puntos. 4. Verificar que el archivo de salida de la nube de puntos se guarde en la ruta indicada en el archivo de configuración. | | |
| Resultado esperado: El programa guarda la nube de puntos de salida en la ruta especificada en el archivo de configuración. | | |
| Resultado obtenido: Una vez realizada la lectura del archivo de configuración y aplicado el algoritmo de apilamiento, se verifica la presencia del archivo de salida en la ruta especificada. <div data-bbox="236 1444 1420 1904" data-label="Image"> </div> | | |

| Caso de prueba 3 | Id. | CP-3 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa utilice las configuraciones del algoritmo indicadas en el archivo de configuración. | | |
| Prerrequisitos Archivo de configuración válido con las configuraciones del algoritmo especificadas. | | |
| <div style="display: flex; justify-content: space-around;"> <pre data-bbox="244 696 826 987"> data > {} config.json > ... 1 { 2 "input_path": "../data/cp-3", 3 "config_params": { 4 "voxel_size": 0.05, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.0123 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 } </pre> <pre data-bbox="847 696 1430 987"> data > {} config.json > output_file 1 { 2 "input_path": "../data/cp-3", 3 "config_params": { 4 "voxel_size": 0.025, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.0123 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 } </pre> </div> | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa lea correctamente el archivo de configuración. 3. Ejecutar el proceso de apilamiento de nubes de puntos utilizando las configuraciones del algoritmo indicadas en el archivo de configuración. 4. Verificar que el comportamiento del programa corresponde a las configuraciones especificadas (por ejemplo, parámetros específicos del algoritmo utilizados durante el procesamiento). | | |
| Resultado esperado: El programa utiliza las configuraciones del algoritmo especificadas en el archivo de configuración para el procesamiento de las nubes de puntos. | | |
| <div style="display: flex; justify-content: space-around;"> <div data-bbox="229 1440 790 1982"> Resultado obtenido: Prueba voxel_size: 0.05:  </div> <div data-bbox="805 1440 1428 1982"> Prueba voxel_size: 0.025:  </div> </div> | | |

| Caso de prueba 4 | Id. | CP-4 |
|------------------|------|----------|
| | Tipo | Unitaria |

Descripción:

Verificar que el programa utilice valores por defecto si el formato de las configuraciones del algoritmo indicadas en el archivo de configuración no es el correcto.

Prerrequisitos

Archivo de configuración con configuraciones del algoritmo en un formato incorrecto.

Prueba 1:

```
data > {} config.json > ...
1 {
2   "input_path": "../data/test copy",
3   "config_params": {
4     "size_of_voxels": "aa",
5     "nb_neighbors": 20,
6     "ratio": "--",
7     "combinability_threshold": 0
8   },
9   "output_file": "../results/nube_pcd_salida.pcd"
10 }
```

Prueba 2:

```
data > {} config.json > ...
1 {
2   "input_path": "../data/test copy",
3   "config_params": {
4     "voxel_size": "aa",
5     "nb_neighbors": 20,
6     "ratio": "--",
7     "combinability_threshold": 0
8   },
9   "output_file": "../results/nube_pcd_salida.pcd"
10 }
```

Pasos:

1. Iniciar la ejecución del programa.
2. Verificar que el programa lea correctamente el archivo de configuración.
3. Verificar que el programa utilice valores por defecto para las configuraciones del algoritmo si el formato es incorrecto.

Resultado esperado:

El programa utiliza valores por defecto para las configuraciones del algoritmo si el formato especificado en el archivo de configuración es incorrecto.

Resultado obtenido:

El programa utiliza valores por defecto para las configuraciones del algoritmo si el formato especificado en el archivo de configuración es incorrecto.

Prueba 1:

```
(venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src$ python3 main.py
Advertencia: Falta la clave 'voxel size' en la configuración.
Advertencia: La configuración no es válida. Se utilizarán los valores por defecto: {'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
{'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
Exito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/test copy'.
Exito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube pcd salida.pcd'.
Se cargaron 4 nubes de puntos.
WARNING: Using soft CircularBuffer (6144 KiB)
FEngine (64 bits) created at 0x601c892d0840 (threading is enabled)
FEngine resolved backend: OpenGL
Advertencia: Falta la clave 'voxel size' en la configuración.
Advertencia: La configuración no es válida. Se utilizarán los valores por defecto: {'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
Los parámetros de configuración son: {'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
```

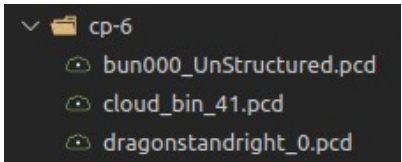
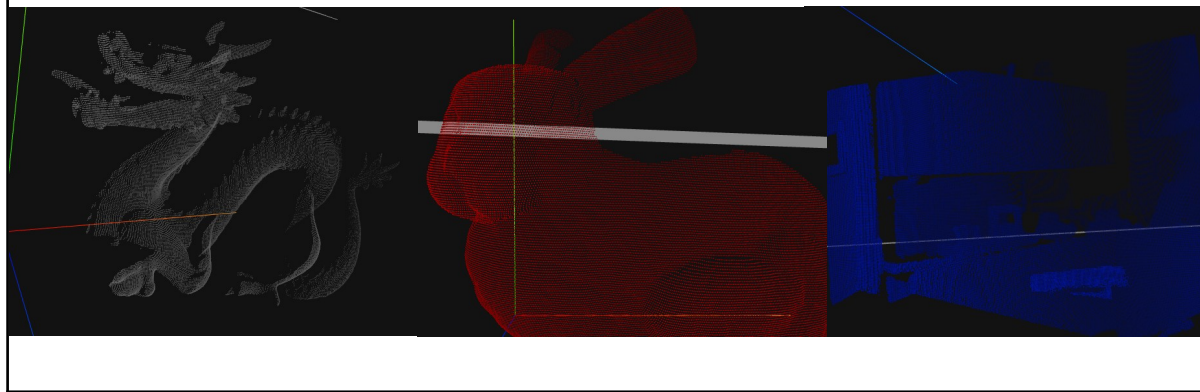
Prueba 2:

```
(venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src$ python3 main.py
Advertencia: El valor de 'voxel size' debe ser de tipo float.
Advertencia: La configuración no es válida. Se utilizarán los valores por defecto: {'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
{'voxel size': 0.02, 'remove outliers params': {'nb_neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.5}
Exito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/test copy'.
Exito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube pcd salida.pcd'.
```

| | | |
|--|--|-----------------------------------|
|  Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 5 | Id. | CP-5 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa muestre un mensaje de error y termine su ejecución si el formato de los archivos de entrada no es el correcto. | | |
| Prerrequisitos Archivos de entrada con un formato incorrecto. <div data-bbox="268 640 978 943"> <pre> data > {} config.json > input_path 1 { 2 "input_path": "../data/cp-3", 3 "config_params": { 4 "voxel_size": 0.02, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.3 10 }, </pre> </div> <div data-bbox="1045 629 1359 943"> <pre> cp-3 ├── cloud_bin_0.ply U ├── cloud_bin_1.ply U ├── cloud_bin_2.ply U ├── cloud_bin_3.ply U ├── cloud_bin_4.ply U ├── cloud_bin_5.ply U └── cloud_bin_6.ply U </pre> </div> | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa detecte el formato incorrecto de los archivos de entrada. 3. Verificar que el programa muestre un mensaje de error y termine su ejecución. | | |
| Resultado esperado: El programa muestra un mensaje de error indicando el formato incorrecto de los archivos de entrada y termina su ejecución. | | |
| Resultado obtenido: El programa muestra un mensaje de error indicando el formato incorrecto de los archivos de entrada y termina su ejecución. <div data-bbox="236 1469 1436 1599"> <pre> • (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src\$ python3 main.py {'voxel size': 0.02, 'remove outliers params': {'nb neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.3} Exito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/cp-3'. Exito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube pc d salida.pcd'. Advertencia: No se encontraron archivos PCD en la carpeta especificada. • (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src\$ </pre> </div> | | |

| | | |
|--|--|-----------------------------------|
| TEC Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 6 | Id. | CP-6 |
|---|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa muestre un mensaje de error y termine su ejecución si las nubes de puntos en los archivos de entrada no tienen suficientes coincidencias según el algoritmo para generar una nube de puntos global. | | |
| Prerrequisitos Archivos de entrada con nubes de puntos que no tienen suficientes coincidencias según el algoritmo. <div data-bbox="261 741 936 1070"> <pre> data > {} config.json > ... 1 { 2 "input_path": "../data/cp-6", 3 "config_params": { 4 "voxel_size": 0.001, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.0123 10 }, 11 "output_file": "../results/nube_pcd_salida_3.pcd" 12 } </pre> </div> <div data-bbox="979 804 1382 965">  </div> | | |
|  | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa detecte que las nubes de puntos en los archivos de entrada no tienen suficientes coincidencias. 3. Verificar que el programa muestre un mensaje de error y termine su ejecución. | | |
| Resultado esperado: El programa muestra un mensaje de error indicando que las nubes de puntos en los archivos de entrada no tienen suficientes coincidencias y termina su ejecución. | | |

Resultado obtenido:

Nubes de puntos sin combinar:




El programa muestra un mensaje de error indicando que las nubes de puntos en los archivos de entrada no tienen suficientes coincidencias y termina su ejecución.

```
• (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src$ python3 main.py
{'voxel size': 0.001, 'remove outliers params': {'nb neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.0123}
Éxito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/cp-6'
Éxito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube_pcd_salida_3.pcd'.
Se cargaron 3 nubes de puntos.
WARNING: Using soft CircularBuffer (6144 KiB)
FEngine (64 bits) created at 0x60fc9bae0a70 (threading is enabled)
FEngine resolved backend: OpenGL
Los parámetros de configuración son: {'voxel size': 0.001, 'remove outliers params': {'nb neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.0123}
Éxito al validar los contenidos del archivo de configuración, iniciando el procesamiento
Remove statistical outliers: [=====] 100%
Remove statistical outliers: [=====] 100%
Remove statistical outliers: [=====] 100%
Se preprocesaron 3 nubes de puntos.
Pair: (PointCloud with 362589 points., PointCloud with 20730 points.), 03D Distance: 2.8641001321130446
Pair: (PointCloud with 362589 points., PointCloud with 21291 points.), 03D Distance: 2.877147406817946
Pair: (PointCloud with 20730 points., PointCloud with 21291 points.), 03D Distance: 0.01981380141895852
Pares combinables: [PointCloud with 362589 points.]
Cantidad de nubes combinables: 1
Las nubes de puntos no tienen suficientes coincidencias para ser combinadas.
• (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src$
```

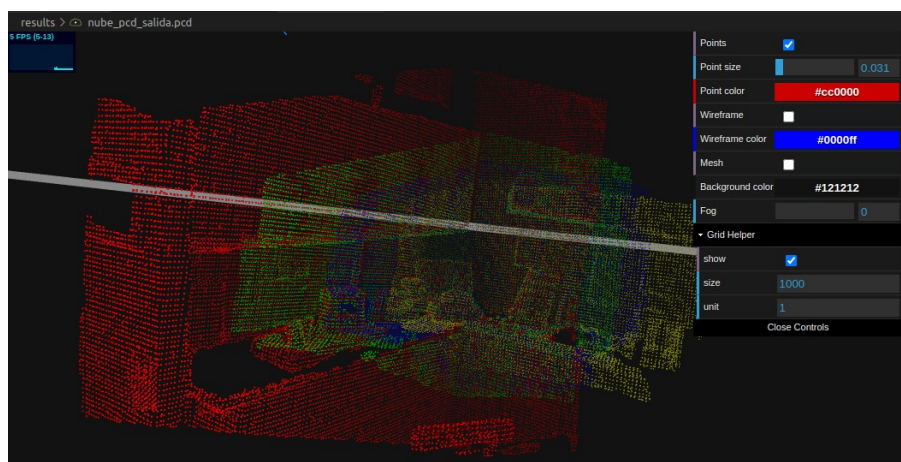
| | | |
|--|--|-----------------------------------|
|  Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 7 | Id. | CP-7 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa muestre un mensaje de éxito al validar los contenidos del archivo de configuración y que indique el inicio del procesamiento. | | |
| Prerrequisitos Archivo de configuración válido. <pre data-bbox="509 658 1160 983"> data > {} config.json > ... 1 { 2 "input_path": "../data/test copy", 3 "config_params": { 4 "voxel_size": 0.02, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.3 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 }</pre> | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa valide correctamente los contenidos del archivo de configuración. 3. Verificar que el programa muestre un mensaje de éxito y que indique el inicio del procesamiento. | | |
| Resultado esperado: El programa muestra un mensaje de éxito al validar los contenidos del archivo de configuración y indica el inicio del procesamiento. | | |
| Resultado obtenido: El programa muestra un mensaje de éxito al validar los contenidos del archivo de configuración y indica el inicio del procesamiento. <pre data-bbox="236 1518 1431 1823"> (venv) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src\$ python3 main.py {'voxel size': 0.02, 'remove outliers params': {'nb neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.3} Exito en la lectura del archivo de configuración: La ruta especificada para nubes de puntos de entrada es: '../data/test copy'. Exito en la lectura del archivo de configuración: La ruta especificada para la nube de puntos de salida es: '../results/nube pcd salida.pcd'. Se cargaron 4 nubes de puntos. WARNING: Using soft CircularBuffer (6144 KiB) FEngine (64 bits) created at 0x55eece861460 (threading is enabled) FEngine resolved backend: OpenGL Los parámetros de configuración son: {'voxel size': 0.02, 'remove outliers params': {'nb neighbors': 20, 'std ratio': 2.0}, 'combinability threshold': 0.3} Exito al validar los contenidos del archivo de configuración, iniciando el procesamiento Remove statistical outliers: [=====] 100% Remove statistical outliers: [=====] 100% Remove statistical outliers: [=====] 100% Remove statistical outliers: [=====] 100% Se preprocesaron 4 nubes de puntos.</pre> | | |

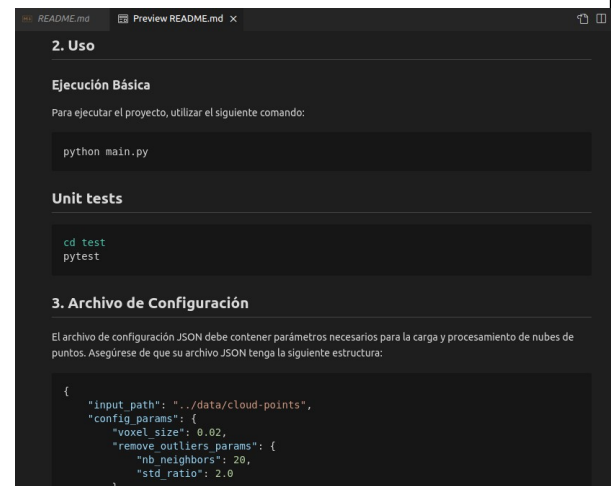
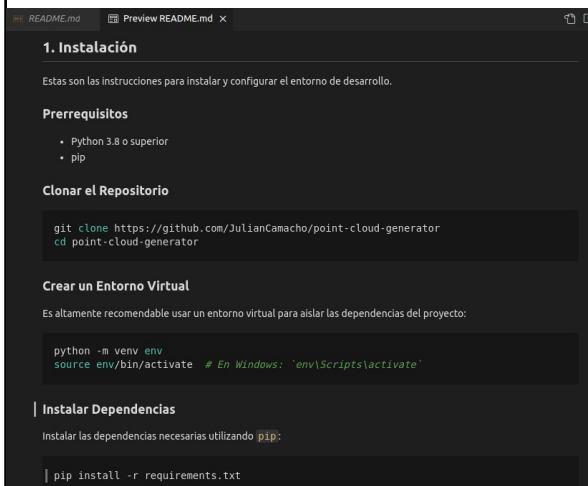
| | | |
|--|--|-----------------------------------|
|  Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 8 | Id. | CP-8 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa muestre un mensaje de éxito al concluir el algoritmo y obtener el archivo de salida. | | |
| Prerrequisitos 1. Archivo de configuración válido. <pre data-bbox="491 658 1142 981"> data > {} config.json > ... 1 { 2 "input_path": "../data/test copy", 3 "config_params": { 4 "voxel_size": 0.02, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.3 10 }, 11 "output_file": "../results/nube_pcd_salida.pcd" 12 }</pre> 2. Nubes de puntos de entrada con suficientes coincidencias según el algoritmo. | | |
| Pasos: <ol style="list-style-type: none"> 1. Iniciar la ejecución del programa. 2. Verificar que el programa aplique correctamente el algoritmo de apilamiento sobre las nubes de puntos de entrada. 3. Verificar que el programa genere el archivo de salida al concluir el algoritmo. 4. Verificar que el programa muestre un mensaje de éxito al finalizar. | | |
| Resultado esperado: El programa muestra un mensaje de éxito al concluir el algoritmo y genera el archivo de salida. | | |
| Resultado obtenido: El programa muestra un mensaje de éxito al concluir el algoritmo y genera el archivo de salida. <pre data-bbox="483 1543 1161 1995"> [Open3D DEBUG] [GlobalOptimizationLM] Optimizing PoseGraph having 4 nodes and 6 edges. [Open3D DEBUG] Line process weight : 9.200700 [Open3D DEBUG] [Initial] residual : 4.266101e+03, lambda : 2.628453e+00 [Open3D DEBUG] [Iteration 00] residual : 1.845110e+01, valid edges : 1, time : 0.000 sec. [Open3D DEBUG] [Iteration 01] residual : 1.842133e+01, valid edges : 1, time : 0.000 sec. [Open3D DEBUG] Current residual - new residual < 1.000000e-06 * current residual [Open3D DEBUG] [GlobalOptimizationLM] total time : 0.000 sec. [Open3D DEBUG] [GlobalOptimizationLM] Optimizing PoseGraph having 4 nodes and 4 edges. [Open3D DEBUG] Line process weight : 12.002700 [Open3D DEBUG] [Initial] residual : 1.298128e-01, lambda : 2.708621e+00 [Open3D DEBUG] [Iteration 00] residual : 1.297924e-01, valid edges : 1, time : 0.000 sec. [Open3D DEBUG] Delta.norm() < 1.000000e-06 * (x.norm() + 1.000000e-06) [Open3D DEBUG] [GlobalOptimizationLM] total time : 0.000 sec. [Open3D DEBUG] CompensateReferencePoseGraphNode : reference : 0 Exito al aplicar el algoritmo, guardando archivo de salida Transformando puntos y combinando en una sola nube de puntos [[1.00000000e+00 1.70994776e-19 0.00000000e+00 0.00000000e+00] [1.70994776e-19 1.00000000e+00 0.00000000e+00 -4.33600869e-19] [-2.16840434e-19 5.42101086e-20 1.00000000e+00 0.00000000e+00] [0.00000000e+00 0.00000000e+00 0.00000000e+00 1.00000000e+00]] [[0.99986257 -0.01650254 0.00158209 0.20618081] [0.01642305 0.99990993 0.04134535 -0.07708214] [-0.00226282 -0.04131369 0.99914366 0.93074099] [0. 0. 0. 1.]] [[0.95986658 -0.01987196 0.27975212 0.18854124] [-0.00846356 0.99497983 0.09971715 -0.11711203] [-0.28032929 -0.09808286 0.9548797 1.31519793] [0. 0. 0. 1.]] [[0.90220919 -0.03739843 0.42967423 0.33191259] [-0.01939245 0.99170843 0.12703667 -0.14201034] [-0.43086253 -0.12294608 0.89400321 1.57447837] [0. 0. 0. 1.]] (env) julian@julian-HP-ProBook-450-G7:~/Documents/github/point-cloud-generator/src\$ </pre> | | |

Archivo de salida:



| Caso de prueba 9 | Id. | CP-9 |
|--|------|----------|
| | Tipo | Unitaria |
| Descripción: Verificar que el programa incluya un archivo README con instrucciones claras sobre el uso del programa, método de ejecución y requisitos del sistema, así como un documento de manual de usuario. | | |
| Prerrequisitos N/A | | |
| Pasos: <ol style="list-style-type: none"> 1. Revisar el contenido del directorio del programa 2. Buscar un archivo README. 3. Verificar que el archivo README contenga instrucciones claras sobre el uso del programa, método de ejecución y requisitos del sistema. 4. Verificar la presencia del manual de usuario. | | |
| Resultado esperado: El programa incluye un archivo README con instrucciones claras sobre el uso del programa, método de ejecución y requisitos del sistema, así como un documento de manual de usuario. | | |
| Resultado obtenido: El programa incluye un archivo README con instrucciones claras sobre el uso del programa, método de ejecución y requisitos del sistema, así como un documento de manual de usuario. | | |



Manual de Usuario

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

AUTOR

José Julián Camacho Hernández
2019201459

ASESOR

Luis Alberto Chavarría Zamora

El presente manual está destinado a proporcionar una guía exhaustiva para el uso del *Point Cloud Generator*. Este *software*, desarrollado como parte del curso CE-5302: Proyecto de Diseño en Ingeniería en Computadores, se enfoca en la generación de mapas tridimensionales a partir de nubes de puntos individuales, sin información de la pose de la cámara. A través de técnicas de visión por computadora y fotogrametría, el proyecto tiene como objetivo principal la construcción de representaciones tridimensionales a partir de datos de entrada de nubes de puntos.

1. Requisitos del Sistema

- Se requiere un sistema operativo compatible con Python 3.8 o superior.
- Acceso a internet para la instalación de dependencias.
- Conexión a una cámara o acceso a archivos de nubes de puntos para procesar.

2. Instalación y Configuración

2.1. Clonar el Repositorio

Para obtener el código fuente del *Point Cloud Generator*, se debe clonar el repositorio desde GitHub utilizando el siguiente comando en la terminal:

```
git clone https://github.com/JulianCamacho/point-cloud-generator
cd point-cloud-generator
```

2.2. Configuración del Entorno Virtual

Se recomienda encarecidamente la creación de un entorno virtual para aislar las dependencias del proyecto. Esto puede lograrse mediante el siguiente comando:

```
python -m venv env
source env/bin/activate # En Windows: 'env\Scripts\activate'
```

2.3. Instalación de Dependencias

Una vez activado el entorno virtual, las dependencias necesarias se pueden instalar ejecutando el siguiente comando:

```
pip install -r requirements.txt
```


3. Configuración del Sistema

El archivo de configuración es un componente crucial del Point Cloud Generator, ya que permite personalizar y controlar los diversos parámetros del proceso de carga, preprocesamiento y combinación de nubes de puntos.

3.1. Estructura del Archivo de Configuración

El archivo de configuración está escrito en formato JSON y debe contener la siguiente estructura:

```
1 {
2   "input_path": "../data/cloud-points",
3   "config_params": {
4     "voxel_size": 0.02,
5     "remove_outliers_params": {
6       "nb_neighbors": 20,
7       "std_ratio": 2.0
8     },
9     "combinability_threshold": 0.5
10  },
11  "output_file": "nube_pcd_combinada.pcd"
12 }
```

3.2. Descripción Detallada de los Parámetros

3.2.1. input_path

Descripción: La ruta al directorio donde se encuentran los archivos de nubes de puntos que se van a procesar.

Ejemplo:

"input_path": "../data/cloud-points"

3.2.2. config_params

Este objeto contiene varios parámetros de configuración que controlan aspectos específicos del procesamiento de las nubes de puntos.

3.2.2.1. voxel_size

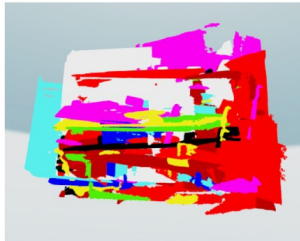
Descripción: Define el tamaño del voxel para el muestreo de las nubes de puntos. El

```
data { | config.json | ...
1 {
2   "input_path": "../data/7-scenes-rookitchen/test",
3   "config_params": {
4     "voxel_size": 0.05,
5     "remove_outliers_params": {
6       "nb_neighbors": 20,
7       "std_ratio": 2.0
8     },
9     "combinability_threshold": 0.17
10  },
11  "output_file": "../results/nube_pcd_solida_5.pcd"
12 }
```

Figura 2: Ejemplo de archivo de configuración.

4.2. Carga de Nubes de Puntos

La aplicación permite cargar nubes de puntos desde archivos locales. Asegúrese de tener los archivos de nubes de puntos en el directorio especificado en el archivo de configuración.



4. Uso de la Aplicación

4.1. Ejecución Básica

Para ejecutar la aplicación, simplemente se debe ejecutar el script main.py desde la terminal:

```
cd src
python3 main.py
```

```
./main.py --input-path ../data/cloud-points --config-param remove_outliers_params.nb_neighbors:20 --std_ratio:2.0 --combinability_threshold:0.50
Exitó en la lectura del archivo de configuración. La ruta especificada para nubes de puntos de entrada es: ../data/7-scenes-rookitchen/test
Exitó en la lectura del archivo de configuración. La ruta especificada para la nube de puntos de salida es: ../results/nube_pcd_solida_5.pcd
Se cargaron 9 nubes de puntos.
Warning: Using safe circularBuffer (4344 KiB)
Pycuda (64 bits) created at 6c4a8f0a2640b (threading is enabled)
Pygame initialized Backend: OpenGL
Pygame initialized Backend: OpenGL
```

Figura 1: Ejecución del programa.

4

```
Aplicando ICP punto a plano (registro fino)...
Construyendo el grafo de poses
Calculando normales para la nube de puntos objetivo...
Aplicando ICP punto a plano (registro grueso)...
Aplicando ICP punto a plano (registro fino)...
Construyendo el grafo de poses
Calculando normales para la nube de puntos objetivo...
Aplicando ICP punto a plano (registro grueso)...
Aplicando ICP punto a plano (registro fino)...
Construyendo el grafo de poses
Optimizando PoseGraph...
OpenSD DEBUG: Validating PoseGraph - finished.
OpenSD DEBUG: GlobalOptimization: Optimizing PoseGraph having 5 nodes and 10 edges.
OpenSD DEBUG: Line process weight: 17.864283
OpenSD DEBUG: [Initial] residual: 6.73937e+04, lambda: 1.466205e+00
OpenSD DEBUG: [Iteration 00] residual: 6.49725e+01, valid edges: 1, time: 0.000 sec.
OpenSD DEBUG: [Iteration 01] residual: 6.48225e+01, valid edges: 1, time: 0.000 sec.
OpenSD DEBUG: [Iteration 02] residual: 6.48206e+01, valid edges: 1, time: 0.000 sec.
OpenSD DEBUG: Delta norm: 1.00000e-06, (x norm): 1.00000e-06
OpenSD DEBUG: [GlobalOptimization] total time: 0.001 sec.
OpenSD DEBUG: [GlobalOptimization] Optimizing PoseGraph having 5 nodes and 5 edges.
OpenSD DEBUG: Line process weight: 14.710500
OpenSD DEBUG: [Initial] residual: 6.35312e-01, lambda: 1.466193e+00
OpenSD DEBUG: [Iteration 00] residual: 6.10611e-01, valid edges: 1, time: 0.000 sec.
OpenSD DEBUG: Delta norm: 1.00000e-06, (x norm): 1.00000e-06
OpenSD DEBUG: [GlobalOptimization] total time: 0.001 sec.
OpenSD DEBUG: CompensationReferencePoseGraph - reference: 0
Exitó al aplicar el algoritmo de combinación de nubes
```

Figura 6: Ejecución del algoritmo de combinación.

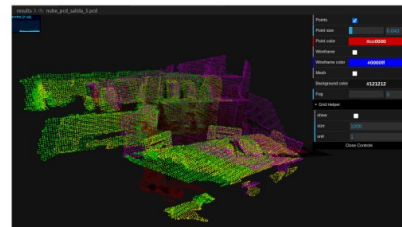
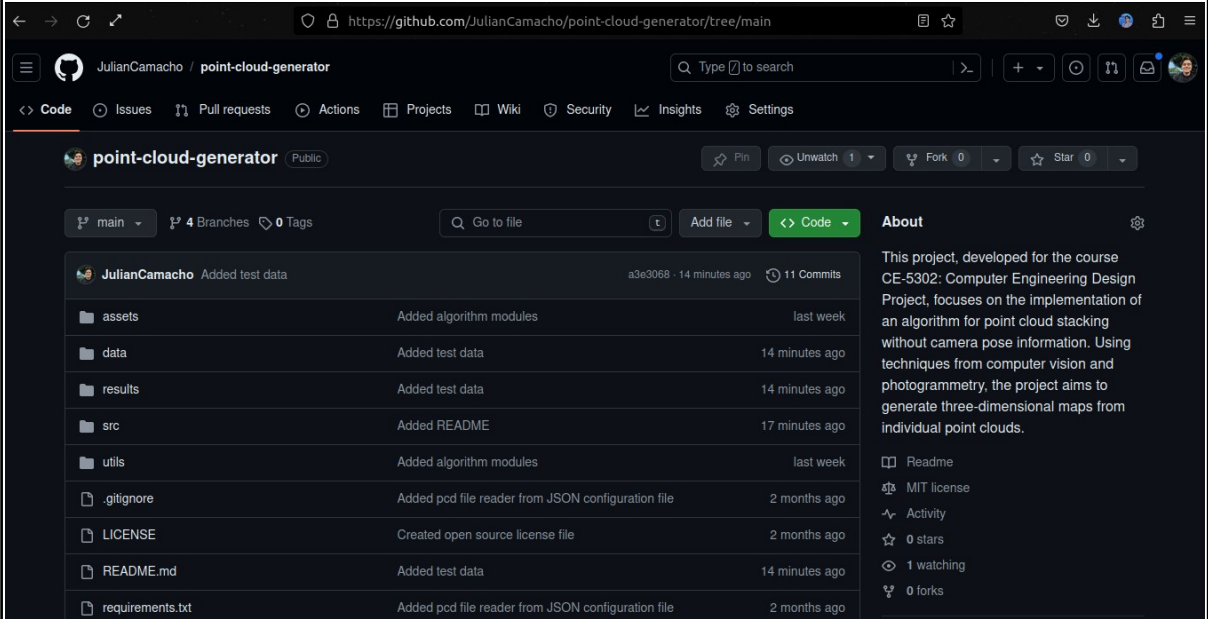


Figura 7: Nube de puntos resultante.

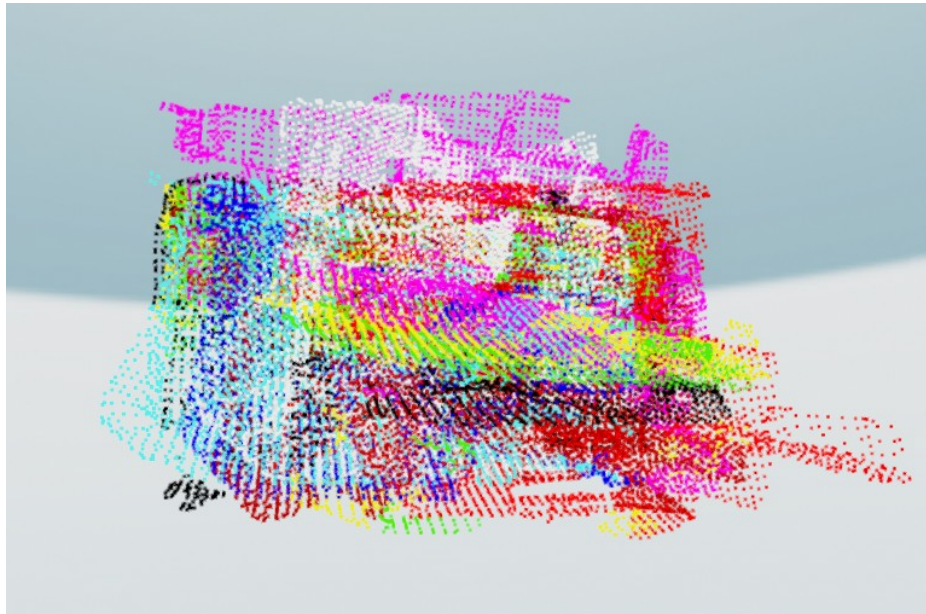
| | | |
|--|--|-----------------------------------|
|  Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--|--|-----------------------------------|

| Caso de prueba 10 | Id. Tipo | CP-10 Unitaria |
|--|-------------|-------------------|
| Descripción: Verificar que el código fuente del programa esté disponible públicamente y sea accesible a través de plataformas de alojamiento de código abierto como GitHub. | | |
| Prerrequisitos N/A | | |
| Pasos: <ol style="list-style-type: none"> 1. Buscar el código fuente del programa en plataformas de alojamiento de código abierto como GitHub. 2. Verificar que el código fuente esté disponible públicamente y sea accesible para su visualización, modificación y redistribución. | | |
| Resultado esperado: El código fuente del programa está disponible públicamente en plataformas de alojamiento de código abierto como GitHub y es accesible para su visualización, modificación y redistribución. | | |
| Resultado obtenido: El código fuente del programa está disponible públicamente en plataformas de alojamiento de código abierto como GitHub y es accesible para su visualización, modificación y redistribución. | | |
|  | | |

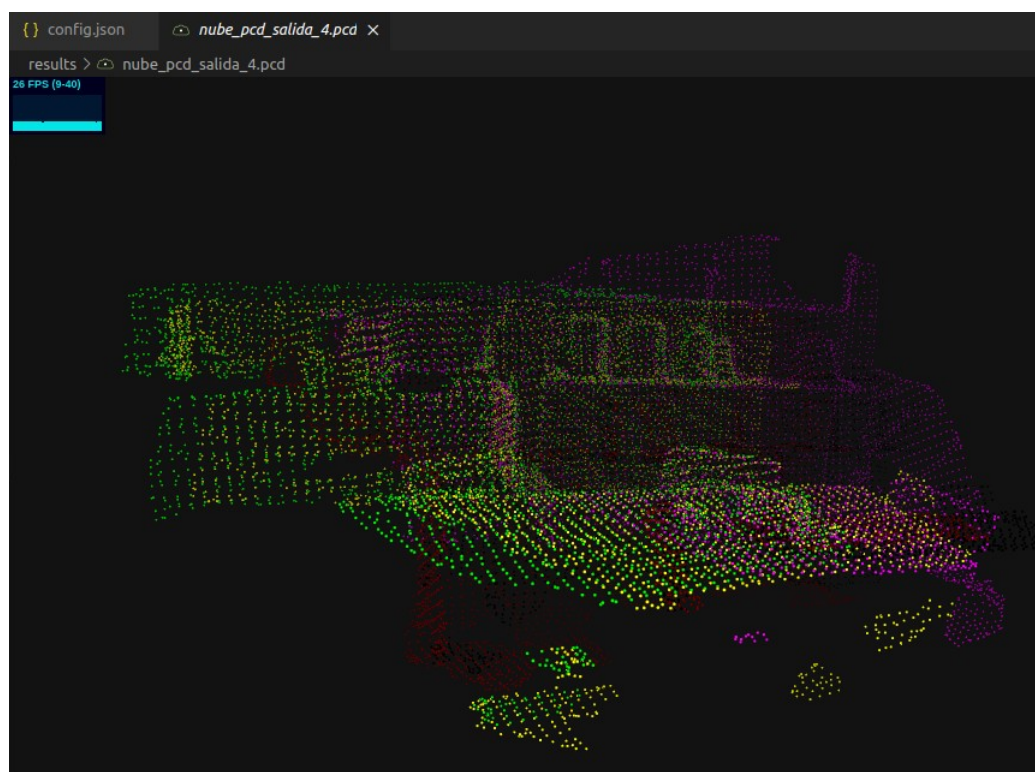
| | | |
|--------------------------------------|--|-----------------------------------|
| TEC Tecnológico de Costa Rica | Generación de nubes de puntos a partir de stacking sin información de pose de la cámara | Ingeniería en Computadores |
|--------------------------------------|--|-----------------------------------|

| Caso de prueba 11 | Id. | CP-11 |
|--|------|-----------|
| | Tipo | Funcional |
| Descripción: Verificar que el programa aplique correctamente el algoritmo de apilamiento sobre las nubes de puntos de entrada y guarde la nube de salida. | | |
| Prerrequisitos: 1. Archivo de configuración válido con las rutas de entrada y salida especificadas. 2. Nubes de puntos de entrada en formato .pcd en las rutas indicadas en el archivo de configuración. <div data-bbox="316 667 938 974" data-label="Text"> <pre> data > {} config.json > ... 1 { 2 "input_path": "../data/7-scenes-redkitchen/test", 3 "config_params": { 4 "voxel_size": 0.05, 5 "remove_outliers_params": { 6 "nb_neighbors": 20, 7 "std_ratio": 2.0 8 }, 9 "combinability_threshold": 0.17 10 }, 11 "output_file": "../results/nube_pcd_salida_4.pcd" 12 } </pre> </div> <div data-bbox="1066 645 1331 981" data-label="Image"> </div> | | |
| Pasos: 1. 1. Iniciar la ejecución del programa. 2. 2. Verificar que el programa aplique el algoritmo de apilamiento sobre las nubes de puntos de entrada. 3. 3. Verificar que el programa guarde la nube de salida en la ruta especificada en el archivo de configuración. | | |
| Resultado esperado: El programa finaliza la ejecución correctamente y se genera la nube de puntos de salida en la ubicación especificada. | | |
| Resultado obtenido: Nubes de puntos de entrada (individuales y sin combinar): <div data-bbox="419 1413 1249 1962" data-label="Image"> </div> | | |

Nubes de puntos de entrada preprocesadas:



La ejecución del programa finaliza correctamente y se genera la nube de puntos de salida con las nubes de puntos compatibles, en la ubicación especificada.



4 REPORTE DE RESULTADOS

Como fue posible verificar en las pruebas, el desarrollo del algoritmo de *stacking* ha arrojado resultados altamente positivos, según las pruebas funcionales y unitarias realizadas. La efectividad del programa se ha visualizado claramente en múltiples pruebas, donde las nubes de puntos se combinaron de manera precisa y coherente.

Las pruebas unitarias demostraron que los componentes individuales del algoritmo funcionan correctamente y cumplen con los requisitos definidos. Los módulos del *software* fueron evaluados para asegurar que se comportara según los requerimientos, validando así la robustez del código y su capacidad para manejar diferentes tipos de datos de entrada.

En las pruebas funcionales, se verificó el comportamiento del sistema completo. Las nubes de puntos generadas a partir de datos fueron procesadas por el algoritmo, y los resultados mostraron una alta coherencia y precisión en la fusión de los puntos. Las evaluaciones visuales confirmaron que el programa logra combinar las nubes de puntos sin información de pose de la cámara de manera efectiva, produciendo mapas tridimensionales que representan fielmente el entorno escaneado.

Los resultados obtenidos demuestran que el proyecto ha alcanzado sus objetivos con éxito. El algoritmo desarrollado es efectivo en la combinación de nubes de puntos, proporcionando una herramienta valiosa para la generación de mapas tridimensionales. Estos resultados positivos no solo validan la viabilidad del enfoque propuesto, sino que también sientan las bases para futuras mejoras y aplicaciones en diversos campos tecnológicos.