Práctica 2

Mapas métricos

Jordi Blasco Lozano

Razonamiento y representación del conocimiento

Grado en Inteligencia Artificial

## Indice:

[Indice: 2](#_Toc186730316)

[1. Objetivo de la práctica 3](#_Toc186730317)

[2. Consideraciones y datos de entrada 3](#_Toc186730318)

[3. Métodos implementados 4](#_Toc186730319)

[Rejilla de Ocupación 4](#_Toc186730320)

[Oc-Tree 4](#_Toc186730321)

[4. Lógica común y detalles de implementación 5](#_Toc186730322)

[Lectura de ficheros PCD 5](#_Toc186730323)

[Estructura de Almacenamiento de Datos 5](#_Toc186730324)

[5. Comparación y cálculo de estadísticas 9](#_Toc186730325)

[6. Parte Optativa: Visualización 3D con PyVista 14](#_Toc186730326)

## Objetivo de la práctica

El objetivo de esta práctica es almacenar y analizar datos métricos en 3D, provenientes de ficheros .pcd (Point Cloud Data), mediante dos estructuras de datos ampliamente usadas en Robótica:

* Rejilla de Ocupación
* Oc-Tree

Ambas estructuras deben:

* Almacenar el número de puntos que caen en cada celda o nodo.
* Calcular la media de las coordenadas (x,y,z) de los puntos que caen en dichas regiones.

El fin último es comparar cuántas celdas o nodos se generan, cuántos quedan vacíos u ocupados y la media de puntos en las celdas/nodos ocupados, para entender mejor las ventajas y desventajas de cada método.

## Consideraciones y datos de entrada

Para la realización de la práctica, contamos con varios **ficheros .pcd** que contienen nubes de puntos 3D en un formato ASCII. Cada uno de estos ficheros incluye:

* Una cabecera con información:
  + FIELDS (x y z r g b)
  + TYPE (F F F I I I)
  + WIDTH, HEIGHT y POINTS (indican la cantidad total de puntos)
* Una sección DATA ascii donde se listan los puntos en formato:  
   “x y z r g b”

en esta practica solo usaremos las tres primeras columnas (x,y,z)

## Métodos implementados

**Rejilla de Ocupación**

La Rejilla de Ocupación realiza una discretización regular del espacio.

* Se define un tamaño de celda, por ejemplo cell\_size = 1.0
* Se calcula un offset con los mínimos (xmin, ymin, Zmin) de la nube de puntos.
* Para cada punto (x,y,z), se mapea a índices de celda (ix,iy,iz).
* Se almacena la suma de coordenadas y el conteo total de puntos en cada índice.

De esta forma, cada celda tendrá:

* **count**: Número de puntos en la celda.
* **mean**: Media de las coordenadas (x,y,z).

## Oc-Tree

La estructura Oc-Tree subdivide recursivamente el espacio en 8 octantes siempre que:

* El número de puntos en un nodo supere un umbral max\_points, y
* El tamaño del nodo (cubo) sea mayor que un tamaño mínimo de celda (min\_cell\_size).

Cada nodo de tipo OctreeNode contiene:

* min\_coord y max\_coord, que definen los límites espaciales de su subcubo.
* Un listado de puntos (en nodos no subdivididos o durante la construcción).
* count y mean (definidos para nodos hoja).
* Una lista de 8 hijos (para los octantes), si se subdivide.

El criterio para detener la subdivisión (marcar nodo como hoja) es cuando el número de puntos no supera max\_points o el tamaño de la región es menor o igual a min\_cell\_size.

## Lógica común y detalles de implementación

## Lectura de ficheros PCD

En ambos métodos, el primer paso es leer las coordenadas 3D de los ficheros .pcd de la siguiente forma:

1. Abrir el fichero.
2. Avanzar línea a línea hasta encontrar DATA ascii.
3. Leer línea a línea cada punto; parsear únicamente (x,y,z).
4. Almacenar estos datos en una lista.

## Estructura de Almacenamiento de Datos

* **Rejilla de Ocupación**

**Paso 1: Identificar los rangos**

Antes de comenzar a asignar puntos a las celdas de la rejilla, es necesario definir bounding box que contiene todos los puntos de la nube. Esto se hace buscando los valores mínimos y máximos de cada eje (x, y, z):

- (x min, x max) Los límites inferior y superior del eje (x).   
- (y min, y max) Los límites inferior y superior del eje (y).   
- (z min, z max) Los límites inferior y superior del eje (z).

Esto define el volumen tridimensional en el que trabajará la rejilla.

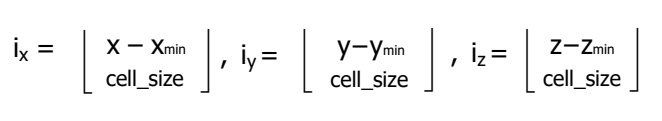
Si por ejemplo tenemos los puntos:

(-5, -7, -3); (4, 2, 6); (-2, -3, 1)

Los límites serían:

x min = -5; x max = 4; y min = -7; y max = 2; z min = -3; z max = 6

**Paso 2: Discretizar cada punto**

Para asignar cada punto a una celda de la rejilla, necesitamos discretizar sus coordenadas continuas (x, y, z) en índices enteros (i\_x, i\_y, i\_z) que identifican una celda específica y esto se hace con la fórmula:

* Resta de desplazamiento: (x - x min): Se traslada la coordenada al origen relativo de la rejilla.
* División por el tamaño de celda: Divide la distancia al origen relativo entre el tamaño de la celda (cell\_size), obteniendo en qué parte de la celda cae el punto.
* Truncaminto: Redondea hacia abajo para asignar un índice entero a la celda correspondiente.

Ejemplo:

A mathematical equation with numbers

Description automatically generatedCon un (cell\_size = 2.0), (x min = -5), y el punto (-3, -6, 0):

El punto se asigna a la celda con índices (i\_x, i\_y, i\_z) = (1, 0, 1)

**Paso 3: Acumular datos en la celda**

Para cada celda (i\_x, i\_y, i\_z), mantenemos dos valores:

count: Número de puntos que caen en la celda.  
sum: La suma acumulada de las coordenadas (x, y, z) de los puntos.

**Paso 4: Calcular la media final**

Para cada celda, calculamos la media de las coordenadas de los puntos almacenados en ella:

Esto nos da la posición promedio de los puntos dentro de esa celda.

* **Oc-Tree**

**Paso 1: Nodo raíz**

El Oc-Tree comienza con un nodo raíz que abarca todo el espacio tridimensional definido por:

min\_coord = [x\_min, y\_min, z\_min]  
max\_coord = [x\_max, y\_max, z\_max]

Este nodo contiene inicialmente todos los puntos de la nube.

**Paso 2: Subdivisión del nodo**

Si el nodo contiene más puntos de los permitidos (max\_points) y su tamaño es mayor al mínimo tamaño de celda permitido (min\_cell\_size), se subdivide en 8 subnodos (octantes).

Cada subnodo corresponde a una subdivisión del espacio en 8 regiones iguales (octantes), definidas por dividir los límites en la mitad:

* El punto medio del nodo para cada uno de los ejes se calcula usando la siguiente fórmula:
* mid\_x = (x\_min + x\_max) / 2
* mid\_y = (y\_min + y\_max) / 2
* mid\_z = (z\_min + z\_max) / 2
* Los octantes tienen los límites:

- **Octante 1**: [x\_min, mid\_x] **x** [y\_min, mid\_y] **x** [z\_min, mid\_z]

- **Octante 2**: Partiendo del Octante 1, vamos cambiando los productos de las multiplicaciones para completar las 8 regiones. Por ejemplo, otro octante podría ser:

[mid\_x, x\_max] **x** [y\_min, mid\_y] **x** [z\_min, mid\_z]

En este caso, hemos tomado la parte del espacio que va de la división del medio hasta el mínimo en los ejes 'y' y 'z' , pero del medio al máximo en el eje 'x'. Esto se hace para completar todas las regiones dividiendo cada eje en dos partes. Finalmente, combinando las divisiones de cada eje, se obtienen las 2^3 regiones tridimensionales es decir los 8 octantes.

**Paso 3: Asignar puntos a los octantes**

Para cada punto del nodo original, verificamos en qué octante cae. Esto se hace comparando las coordenadas del punto con los límites de los octantes.

Ejemplo:

Supongamos que tenemos un punto p = (xp, yp, zp). Lo asignamos a un octante verificando:

- Si xmin ≤ xp < midx: Pertenece a la parte izquierda del eje x.  
- Si ymin ≤ yp < midy: Pertenece a la parte inferior del eje y.  
- Si zmin ≤ zp < midz: Pertenece a la parte frontal del eje z.

Repitiendo esta lógica, encontramos exactamente en qué octante debe estar el punto.

Este proceso es recursivo: cada subnodo puede subdividirse nuevamente en 8 octantes nuevos si cumple las condiciones de subdivisión anteriormente mencionadas, si esto pasara el subnodo entraría al paso 2 nuevamente, en caso contrario pasaría al paso 4

**Paso 4: Nodos hoja**

Cuando un nodo ya no se puede subdividir por tener menos demax\_points o alcanzar min\_cell\_size, se convierte en un nodo hoja.

En los nodos hoja:

- Se almacena el número de puntos (count).  
- Se calcula la media de las coordenadas de los puntos almacenados (mean).

* **Diferencias clave**

|  |  |  |
| --- | --- | --- |
| Característica | Rejilla de Ocupación | Oc-Tree |
| Resolución | Fija | Adaptativa |
| Estructura de datos | Matriz o diccionario | Árbol jerárquico |
| Eficiencia espacial | Consume más espacio para zonas vacías | Más eficiente (menos nodos en zonas vacías) |
| Complejidad | |  | | --- | | Simplicidad de implementación |  |  | | --- | |  | | Más compleja (recursión) |

## Comparación y cálculo de estadísticas

En ambos métodos se implementa una función que recorre la estructura y obtiene las siguientes estadisticas

**Rejilla:**

* total\_celdas: número de claves del diccionario
* ocupadas: cuántas celdas tienen count > 0.
* vacias: celdas que no llegaron a usarse (si se instancia todo el volumen) o 0 si solo se crea celda cuando hay puntos.
* media\_puntos\_ocupadas: la media del count en las celdas ocupadas.

**Oc-tree:**

* total\_nodos: número total de nodos en el árbol.
* hojas: cantidad de nodos sin hijos (is\_leaf=True).
* internas: cantidad de nodos con hijos (is\_leaf=False).
* ocupadas: hojas con count > 0.
* vacias: hojas con count = 0.
* media\_puntos\_ocupadas: media del count en hojas ocupadas.

Debemos de evaluar estos datos usando diferentes valores para los parametros de ***cell\_size*** para la regilla de ocupación y de ***min\_cell\_octree*** y ***max\_points*** para el Oc-Tree. Una vez obtenidos los resultados de todos los archivos .pcd con distintos valores para los parametros podemos obtener las siguientes observaciones y conclusiones.

**Resultados**

1. Con (cell\_size=1.0, min\_cell\_octree=1.0, max\_points=100)
2. Con (cell\_size=0.5, min\_cell\_octree=0.5, max\_points=100)
3. Con (cell\_size=3.0, min\_cell\_octree=1.0, max\_points=300)

Analizando: ciencias000

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias000.pcd

Rejilla de Ocupacion:

Total de Celdas: 3255

Celdas Ocupadas: 450

Celdas Vacias: 2805

Media de puntos en celdas ocupadas: 130.72

Oc-Tree:

Total de Nodos: 929

Hojas: 813

Nodos Internos: 116

Celdas (Hojas) Ocupadas: 435

Celdas (Hojas) Vacias: 378

Media de puntos en hojas ocupadas: 135.23

-------------------

-------------------

Analizando: ciencias001

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias001.pcd

Rejilla de Ocupacion:

Total de Celdas: 3780

Celdas Ocupadas: 447

Celdas Vacias: 3333

Media de puntos en celdas ocupadas: 129.52

Oc-Tree:

Total de Nodos: 937

Hojas: 820

Nodos Internos: 117

Celdas (Hojas) Ocupadas: 440

Celdas (Hojas) Vacias: 380

Media de puntos en hojas ocupadas: 131.55

-------------------

-------------------

Analizando: scan000

Analisis Comparativo

-------------------

Archivo: ./Datos/scan000.pcd

Rejilla de Ocupacion:

Total de Celdas: 568524

Celdas Ocupadas: 4262

Celdas Vacias: 564262

Media de puntos en celdas ocupadas: 189.43

Oc-Tree:

Total de Nodos: 15825

Hojas: 13847

Nodos Internos: 1978

Celdas (Hojas) Ocupadas: 7149

Celdas (Hojas) Vacias: 6698

Media de puntos en hojas ocupadas: 112.93

-------------------

-------------------

Analizando: museo000

Analisis Comparativo

-------------------

Archivo: ./Datos/museo000.pcd

Rejilla de Ocupacion:

Total de Celdas: 192

Celdas Ocupadas: 50

Celdas Vacias: 142

Media de puntos en celdas ocupadas: 1582.98

Oc-Tree:

Total de Nodos: 1393

Hojas: 1219

Nodos Internos: 174

Celdas (Hojas) Ocupadas: 591

Celdas (Hojas) Vacias: 628

Media de puntos en hojas ocupadas: 133.90

-------------------

-------------------

Analizando: poli000

Analisis Comparativo

-------------------

Archivo: ./Datos/poli000.pcd

Rejilla de Ocupacion:

Total de Celdas: 1980

Celdas Ocupadas: 251

Celdas Vacias: 1729

Media de puntos en celdas ocupadas: 250.29

Oc-Tree:

Total de Nodos: 1097

Hojas: 960

Nodos Internos: 137

Celdas (Hojas) Ocupadas: 568

Celdas (Hojas) Vacias: 392

Media de puntos en hojas ocupadas: 110.60

-------------------

-------------------

Analizando: poli001

Analisis Comparativo

-------------------

Archivo: ./Datos/poli001.pcd

Rejilla de Ocupacion:

Total de Celdas: 1836

Celdas Ocupadas: 232

Celdas Vacias: 1604

Media de puntos en celdas ocupadas: 264.78

Oc-Tree:

Total de Nodos: 945

Hojas: 827

Nodos Internos: 118

Celdas (Hojas) Ocupadas: 561

Celdas (Hojas) Vacias: 266

Media de puntos en hojas ocupadas: 109.48

-------------------

-------------------

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias000.pcd

Rejilla de Ocupacion:

Total de Celdas: 617136

Celdas Ocupadas: 6563

Celdas Vacias: 610573

Media de puntos en celdas ocupadas: 8.96

Oc-Tree:

Total de Nodos: 2945

Hojas: 2577

Nodos Internos: 368

Celdas (Hojas) Ocupadas: 1413

Celdas (Hojas) Vacias: 1164

Media de puntos en hojas ocupadas: 41.63

-------------------

-------------------

Analizando: ciencias001

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias001.pcd

Rejilla de Ocupacion:

Total de Celdas: 768060

Celdas Ocupadas: 6436

Celdas Vacias: 761624

Media de puntos en celdas ocupadas: 9.00

Oc-Tree:

Total de Nodos: 2889

Hojas: 2528

Nodos Internos: 361

Celdas (Hojas) Ocupadas: 1365

Celdas (Hojas) Vacias: 1163

Media de puntos en hojas ocupadas: 42.40

-------------------

-------------------

Analizando: scan000

Analisis Comparativo

-------------------

Archivo: ./Datos/scan000.pcd

Rejilla de Ocupacion:

Total de Celdas: 122067792

Celdas Ocupadas: 72292

Celdas Vacias: 121995500

Media de puntos en celdas ocupadas: 11.17

Oc-Tree:

Total de Nodos: 48849

Hojas: 42743

Nodos Internos: 6106

Celdas (Hojas) Ocupadas: 21582

Celdas (Hojas) Vacias: 21161

Media de puntos en hojas ocupadas: 37.41

-------------------

-------------------

Analizando: museo000

Analisis Comparativo

-------------------

Archivo: ./Datos/museo000.pcd

Rejilla de Ocupacion:

Total de Celdas: 23465

Celdas Ocupadas: 1192

Celdas Vacias: 22273

Media de puntos en celdas ocupadas: 66.40

Oc-Tree:

Total de Nodos: 4033

Hojas: 3529

Nodos Internos: 504

Celdas (Hojas) Ocupadas: 1735

Celdas (Hojas) Vacias: 1794

Media de puntos en hojas ocupadas: 45.61

-------------------

-------------------

Analizando: poli000

Analisis Comparativo

-------------------

Archivo: ./Datos/poli000.pcd

Rejilla de Ocupacion:

Total de Celdas: 351000

Celdas Ocupadas: 4898

Celdas Vacias: 346102

Media de puntos en celdas ocupadas: 12.83

Oc-Tree:

Total de Nodos: 3249

Hojas: 2843

Nodos Internos: 406

Celdas (Hojas) Ocupadas: 1684

Celdas (Hojas) Vacias: 1159

Media de puntos en hojas ocupadas: 37.30

-------------------

-------------------

Analizando: poli001

Analisis Comparativo

-------------------

Archivo: ./Datos/poli001.pcd

Rejilla de Ocupacion:

Total de Celdas: 371680

Celdas Ocupadas: 4775

Celdas Vacias: 366905

Media de puntos en celdas ocupadas: 12.86

Oc-Tree:

Total de Nodos: 2937

Hojas: 2570

Nodos Internos: 367

Celdas (Hojas) Ocupadas: 1560

Celdas (Hojas) Vacias: 1010

Media de puntos en hojas ocupadas: 39.37

-------------------

-------------------

Analizando: ciencias000

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias000.pcd

Rejilla de Ocupacion:

Total de Celdas: 79120

Celdas Ocupadas: 2477

Celdas Vacias: 76643

Media de puntos en celdas ocupadas: 23.75

Oc-Tree:

Total de Nodos: 2521

Hojas: 2206

Nodos Internos: 315

Celdas (Hojas) Ocupadas: 1223

Celdas (Hojas) Vacias: 983

Media de puntos en hojas ocupadas: 48.10

-------------------

-------------------

Analizando: ciencias001

Analisis Comparativo

-------------------

Archivo: ./Datos/ciencias001.pcd

Rejilla de Ocupacion:

Total de Celdas: 97524

Celdas Ocupadas: 2361

Celdas Vacias: 95163

Media de puntos en celdas ocupadas: 24.52

Oc-Tree:

Total de Nodos: 2433

Hojas: 2129

Nodos Internos: 304

Celdas (Hojas) Ocupadas: 1128

Celdas (Hojas) Vacias: 1001

Media de puntos en hojas ocupadas: 51.31

-------------------

-------------------

Analizando: scan000

Analisis Comparativo

-------------------

Archivo: ./Datos/scan000.pcd

Rejilla de Ocupacion:

Total de Celdas: 15280056

Celdas Ocupadas: 23941

Celdas Vacias: 15256115

Media de puntos en celdas ocupadas: 33.72

Oc-Tree:

Total de Nodos: 48849

Hojas: 42743

Nodos Internos: 6106

Celdas (Hojas) Ocupadas: 21582

Celdas (Hojas) Vacias: 21161

Media de puntos en hojas ocupadas: 37.41

-------------------

-------------------

Analizando: museo000

Analisis Comparativo

-------------------

Archivo: ./Datos/museo000.pcd

Rejilla de Ocupacion:

Total de Celdas: 3360

Celdas Ocupadas: 338

Celdas Vacias: 3022

Media de puntos en celdas ocupadas: 234.17

Oc-Tree:

Total de Nodos: 2793

Hojas: 2444

Nodos Internos: 349

Celdas (Hojas) Ocupadas: 1212

Celdas (Hojas) Vacias: 1232

Media de puntos en hojas ocupadas: 65.29

-------------------

-------------------

Analizando: poli000

Analisis Comparativo

-------------------

Archivo: ./Datos/poli000.pcd

Rejilla de Ocupacion:

Total de Celdas: 45630

Celdas Ocupadas: 1625

Celdas Vacias: 44005

Media de puntos en celdas ocupadas: 38.66

Oc-Tree:

Total de Nodos: 2777

Hojas: 2430

Nodos Internos: 347

Celdas (Hojas) Ocupadas: 1437

Celdas (Hojas) Vacias: 993

Media de puntos en hojas ocupadas: 43.71

-------------------

-------------------

Analizando: poli001

Analisis Comparativo

-------------------

Archivo: ./Datos/poli001.pcd

Rejilla de Ocupacion:

Total de Celdas: 48960

Celdas Ocupadas: 1654

Celdas Vacias: 47306

Media de puntos en celdas ocupadas: 37.14

Oc-Tree:

Total de Nodos: 2473

Hojas: 2164

Nodos Internos: 309

Celdas (Hojas) Ocupadas: 1373

Celdas (Hojas) Vacias: 791

Media de puntos en hojas ocupadas: 44.73

-------------------

-------------------

**Observaciones**A partir de los resultados obtenidos con las configuraciones de parámetros:

**1) (cell\_size=1.0, min\_cell\_octree=1.0, max\_points=100):**

**Regilla de Ocupación**

- Los archivos analizados muestran un número total de celdas que oscila, por ejemplo, entre 3360 (museo000) y 15 280 056 (scan000).  
- En la mayoría de los casos, las celdas vacías superan el 90% del total (por ejemplo, en ciencias000, 76 643 de 79 120 celdas están vacías ≈ 97%; en scan000, 15 256 115 de 15 280 056≈99%)  
- Esto sucede porque el espacio se divide uniformemente, pero solo una pequeña fracción de las celdas contiene puntos.  
- La media de puntos por celda ocupada varía bastante según el archivo (desde cerca de 23–24 en ciencias000 y ciencias001, hasta 234 en museo000), lo que indica que algunas regiones concentran muchos puntos.

**Oc-Tree**

- El número total de nodos abarca desde valores relativamente pequeños (2473 en poli001) hasta decenas de miles (48849 en scan000).  
- La proporción de hojas ocupadas frente a hojas totales está en torno al 50–60% en varios casos (por ejemplo, en ciencias000 hay 1223 hojas ocupadas de 2206 ≈ 55%).  
- La media de puntos en las hojas ocupadas (48.10 en ciencias000, 65.29 en museo000) suele ser superior a la de la rejilla, lo que indica una agrupación más eficiente de puntos en los nodos.

En resumen, con estas configuraciones (celda = 1.0), la rejilla genera muchas celdas vacías, mientras que el Oc-Tree logra una ocupación más concentrada en las regiones de interés.

**2) (cell\_size=0.5, min\_cell\_octree=0.5, max\_points=100):**

**Regilla de Ocupación**

- Reducir el tamaño de las celdas de la rejilla incrementa drásticamente el número total de celdas (hasta 617 136). Aunque esto mejora la resolución de la rejilla, también aumenta el número de celdas vacías.  
- Consecuentemente, también aumenta el número (y el porcentaje) de celdas vacías hasta un 98%   
- A pesar de tener mejor resolución, esto implica almacenar más celdas sin datos.

**Oc-Tree**

- Se incrementa el número total de nodos y hojas con respecto a la configuración anterior, porque ahora el árbol se subdivide más para ajustarse al tamaño de celda más pequeño.  
- Sin embargo, sigue siendo mucho más compacto que la rejilla: las hojas vacías son menos de la mitad en muchos casos, y la media de puntos en hojas ocupadas (entre 37 y 51 en los distintos archivos)

Con un cell\_size de 0.5 se incrementa exponencialmente las celdas vacías en la rejilla, mientras que el Oc-Tree continúa mostrando ventajas en la distribución de nodos.

**3) (cell\_size=3.0, min\_cell\_octree=1.0, max\_points=300):**

**Regilla de Ocupación**

- Aumentar el tamaño de la celda a 3.0 reduce drásticamente el número total de celdas, a su vez se reduce tambien el porcentaje de celdas vacias hasta un 86% - La media de puntos en las celdas ocupadas aumenta ya que las celdas abarcan mas volumen y consiguen retener mas puntos

**Oc-Tree**

**-** Se ve un descenso en el número total de nodos y de hojas en comparación con configuraciones más finas, ya que se realizan menos subdivisiones- La media de puntos por hoja ocupada aumenta notablemente reflejando que cada hoja cubre un volumen mayor con más puntos acumulados.

Esta configuración de celdas grandes reduce drásticamente la cantidad de nodos totales tanto en la rejilla como en el Oc-Tree, a costa de menor precisión. Aun así, el Oc-Tree continúa beneficiándose de una subdivisión adaptativa en las regiones más densas.

**Comparación General**

**Rejilla de Ocupación**

- Es simple de implementar y adecuada para entornos donde se requiere una división uniforme del espacio.  
- Sufre de ineficiencia en regiones con muchos puntos vacíos, especialmente con tamaños de celda pequeños.  
- Aumentar el tamaño de celda mejora la eficiencia, pero reduce la resolución espacial.

**Oc-Tree**

- Es más eficiente para representar datos en 3D, adaptándose a las regiones densas y dejando sin dividir las áreas vacías.  
- Produce menos nodos totales en comparación con las celdas de la rejilla.  
- Permite un balance entre resolución y eficiencia al ajustar los parámetros min\_cell\_octree y max\_points.

**Conclusión Final**

El Oc-Tree es claramente superior a la rejilla de ocupación en términos de eficiencia y adaptabilidad para datos tridimensionales dispersos. Mientras que la rejilla de ocupación es útil para representaciones uniformes, el Oc-Tree ofrece una solución más compacta y ajustada a la distribución de puntos, lo que lo hace más adecuado para aplicaciones que requieren almacenamiento eficiente y resolución dinámica.

## Parte Optativa: Visualización 3D con PyVista

Consiste en la visualización 3D de los datos y la estructura del Oc-Tree , utilizando la librería PyVista. El visualizador permite observar tanto los puntos originales de la nube como las celdas generadas por el Oc-Tree.

**Implementación:**

- Se utiliza la librería PyVista para renderizar la nube.

- Cada fichero .pcd se lee completamente y se construye un objeto PolyData que contiene todos los puntos.

- Se crea un plot con el objeto, permite rotar, hacer zoom y desplazarse por el espacio 3D

A computer screen shot of a building

Description automatically generated- En la clase Oc-tree se ha implementado un metodo que consiste en guardar una lista de todos los nodos que genera el Oc-Tree subdividiendose, para exportarlos y representarlos en formato de cubos en el plot junto a los puntos de los archivos .pcd correspondiente