Reconstrucción 3D de objetos

Carlos Garay

Resumen– Con el transcurso de los años ha ido surgiendo la necesidad de hacer uso de la reconstrucción 3D (pasar elementos del mundo real a un entorno digital). Podemos ver como en diferentes ámbitos se hace uso de la reconstrucción 3D. Por ejemplo, en la industria de los videojuegos para crear mundos virtuales más realistas, en el cine para perfeccionar el CGI (*Computer Generated Imagery*), en la medicina para la generación de prótesis o en el sector automovilístico para integrar tecnologías de autoconducción.
Con el avance de la tecnología, las técnicas de reconstrucción 3D han ido evolucionando. Uno de los fines de este estudio es ver las diferentes alternativas que tenemos actualmente en nuestra disposición. En este trabajo se realizarán diferentes implementaciones, entre ellas tenemos el uso de una cámara estereoscópica *Intel Realsense D435* [5], una implementación simple de visión estéreo con la librería *OpenCV* [15][8] y también implementaciones más complejas de fotogrametría con métodos más clásicos SfM *(Structure from Motion)* [22] y métodos más innovadores como el uso de aprendizaje computacional NeRF *(Neural Radiance Fields)*[21][4].
Palabras clave Reconstrucción 3D, visión estéreo, calibración, geometría epipolar, cámara estereoscópica, SfM, Intel Realsense, aprendizaje computacional, NeRF.

Abstract– Over the years, there has been an increase in the need to make use of 3D reconstruction. Transferring elements from the real world to digital environments is starting to become essential in various different fields. For instance, creating more realistic virtual worlds in the video game industry, perfecting CGI (Computer Generated Imagery) for the cinematic world, generating better quality prosthetics in the medical field, or even integrating self-driving vehicles with new technology in the automotive industry.
With the advancement of technology, 3D reconstruction techniques have been evolving. One of the purposes of this study is to see the different alternatives that society currently has at disposal. In this work different implementations will be performed, among them we have the use of a stereoscopic camera *Intel Realsense D435* [5], a simple implementation of stereo vision with the *OpenCV* [15][8] library and also more complex implementations of photogrammetry with more classical methods SfM (Structure from Motion) [22] and more innovative methods such as the use of computational learning NeRF (Neural Radiance Fields)[21][4].

Keywords– 3D reconstruction, stereo vision, calibration, epipolar geometry, stereo-optical camera, Structure from Motion, Intel Realsense, machine learning, Neural Radiance Fields.

1 Introducción - Contexto del trabajo

La reconstrucción 3D es el proceso mediante el cual elementos del mundo real son reproducidos en un entorno digital, manteniendo sus características físicas (dimensiones, volumen y forma). Podemos encontrar diferentes aplicaciones para la reconstrucción 3D:

- Entretenimiento: Los escáneres 3D son utilizados por la industria del entretenimiento para crear modelos digitales en películas y videojuegos. Suele ser mucho más rápido escanear el objeto del mundo real, que el artista tenga que crearlo manualmente con un software de modelado 3D.

- Bienes y raíces: Los terrenos o edificios se pueden escanear en un modelo 3D, lo que permite a los compradores recorrerlos e inspeccionarlos de forma remota, sin tener que estar presentes en la propiedad.

• E-mail de contacte: 1530356@uab.cat
• Mención realizada: Computación
• Trabajo tutorizado por: Coen Antens (Centro Visión por Computador)
• Curs 2021/22

- Patrimonio cultural: El uso combinado de tecnologías de escaneo 3D e impresión 3D permite replicar artefactos delicados para su posterior estudio, con un método poco invasivo.
- Médico: Los escáneres 3D se utilizan en ortopedia y odontología. Para diseñar y fabricar las órtesis, prótesis o implantes de un paciente.
- Fines industriales: En este entorno el escaneo 3D puede tener diferentes fines como el control de calidad, determinar patrones de desgaste, analizar la superficie de componentes complejos, realizar ingeniería inversa de componentes, etc.

2 OBJETIVOS

El objetivo de TFG es ver y comprender los conceptos básicos que envuelven a la reconstrucción 3D y realizar una comparativa de diferentes métodos que tenemos en nuestra disposición. Concretamente, se quieren alcanzar los siguientes objetivos:

- Cimentar las bases de la reconstrucción 3D, por esto al principio se realiza un estudio más teórico, esto se debe a que los métodos de reconstrucción que se han realizado tienen un proceso de implementación relativamente sencillo, pero tienen un trasfondo mucho más complejo.
- Realizar la reconstrucción 3D con una cámara estereoscópica denominada *Intel Realsense D435* [5].
- Realizar la reconstrucción 3D haciendo uso de un algoritmo clásico SfM *(Structure from Motion)* [22].
- Realizar la reconstrucción 3D haciendo de un modelo de aprendizaje computacional NeRF (*Neural Radiance Fields*)[21][4],

3 METODOLOGÍA

Para cumplir los objetivos descritos anteriormente, se ha decidido desarrollar el proyecto en 4 fases. En la primera, realizar una investigación más teórica y profundizar en las bases de la reconstrucción 3D. En la segunda, realizar una reconstrucción a partir de una cámara estereoscópica [5]. La tercera realizar el uso de un algoritmo clásico de reconstrucción. Por último, implementar un modelo de aprendizaje computacional.
Para asegurar el buen desarrollo del proyecto, se han hecho reuniones, prácticamente semanales, con el tutor del proyecto. En estas se ha hablado de posibles modificaciones, correcciones a realizar por el trabajo hecho previamente y se han resuelto dudas.

3.1 Primera fase

En esta primera fase, aparte del estudio más teórico, también se han realizado pequeñas implementaciones sobre conceptos básicos de visión estéreo[20]. Estas implementaciones se han realizado con el lenguaje Python, haciendo uso del *IDE PyCharm* y principalmente de la librería *OpenCV* [15][8].

3.2 Segunda fase

Para realizar la reconstrucción 3D, se hizo uso del dispositivo *Intel Realsense D435*[5], que es una cámara estereoscópica. Para poder hacer uso del dispositivo se empleó el software específico *Intel Realsense SDK 2.0* y posteriormente el programa *CloudCompara*[1] para realizar el postprocesador.

3.3 Tercera fase

En este punto, para realizar la reconstrucción 3D, se empleó una técnica de fotogrametría. Se hizo uso del software *MeshRoom*[6] que implementa un método clásico de reconstrucción SfM [22].

3.4 Cuarta fase

Por último, se puso en marcha un modelo de aprendizaje computacional NeRF [21][4]. Para poder hacer uso del modelo, era necesario compilarlo en una máquina que tenía una serie de requisitos (una *GPU NVIDIA*, un compilador compatible con *C++14*, *CUDA v10.2* o superior, *CMake v3.21* o superior, (opcional) *Python 3.7* o superior, (opcional) *OptiX 7.3* o superior).

4 ESTADO DEL ARTE

Existen distintas propuestas para el proceso de reconstrucción de objetos 3D. Se pueden dividir en reconstrucción de contacto y sin contacto. A su vez, las soluciones sin contacto se pueden dividir en activas y pasivas [10].
Las soluciones por contacto suelen ser mucho más precisas que los métodos sin contacto, pero estas suelen requerir mucho más tiempo para realizar el escaneo de un objeto. Por otra parte, las soluciones activas sin contacto, utilizan un hardware específico que emite algún tipo de radiación como puede ser rayos X, láseres, infrarrojos, etc. En contraposición, los métodos pasivos sin contactos suelen ser más económicos, porque en la mayoría de los casos no necesitan un hardware particular, sino cámaras digitales simples.

4.1 Por contacto

Los escáneres por contacto sondean al sujeto a través del tacto físico. Una *CMM* (máquina de medición de coordenadas) [2] es un ejemplo de un escáner 3D de contacto. En las *CMM* se utilizan varios tipos de sondas, incluidas las mecánicas, ópticas, láser y de luz blanca. Las *CMM* permiten el movimiento de la sonda a lo largo de los ejes X, Y y Z, muchas máquinas también permiten controlar el ángulo de la sonda. Cada eje tiene un sensor de gran precisión, de esta manera cuando la sonda entra en contacto es capaz de medir la posición de un punto en la superficie del objeto.

Se utiliza principalmente en el proceso de fabricación de piezas y puede ser muy preciso. Sin embargo, la desventaja de las *CMM* es que requieren contacto con el objeto que se está escaneando. Por lo tanto, el acto de escanear el objeto podría modificarlo o dañarlo. La otra desventaja de las *CMM* es que son relativamente lentas en comparación con los otros métodos de escaneo.

4.2 Activo sin contacto

Los escáneres activos sin contacto para sondear, tienen un dispositivo específico que emiten algún tipo de radiación y miden el reflejo que se produce en un objeto. Los posibles tipos de emisiones utilizados incluyen luz, ultrasonido o rayos X. A continuación se enumerarán algunos métodos de reconstrucción.

- Tiempo de vuelo [2]: es un escáner activo que utiliza luz láser para sondear al sujeto. Este tipo de escáner, utiliza un telémetro láser que emite un pulso de luz y mide el tiempo de ida y vuelta para determinar la distancia.
  El telémetro láser solo detecta la distancia de un punto en su dirección de visión. Para escanear más puntos se gira el telémetro o en su lugar se utiliza un sistema de espejos giratorios que suele ser más común, ya que los espejos son más ligeros y fáciles de manejar.
  La ventaja de los telémetros de tiempo de vuelo es que son capaces de operar en distancias muy largas, del orden de kilómetros. Por lo tanto, estos escáneres son adecuados para escanear grandes estructuras como edificios o características geográficas. La desventaja de los telémetros de tiempo de vuelo es su precisión. Debido a la alta velocidad de la luz, la sincronización del tiempo de ida y vuelta es difícil de medir y la precisión es relativamente baja, del orden de milímetros.

- Triangulación [2]: Con respecto al escáner láser 3D de tiempo de vuelo, este proyecta un láser sobre el objeto y utiliza una cámara para saber el punto de impacto. Sabiendo el ángulo y la distancia entre el emisor láser y la cámara, se puede ubicar el objeto en un espacio 3D. Tienen un alcance limitado de algunos metros, pero su precisión es relativamente alta. La precisión de los telémetros de triangulación es del orden de decenas de micrómetros.

- Luz estructurada [2]: Los escáneres 3D de luz estructurada proyectan un patrón de luz sobre el sujeto y observan la deformación del patrón en el sujeto. El patrón se proyecta sobre el sujeto utilizando un proyector LCD u otra fuente de luz estable. Una cámara, ligeramente desplazada, observa la forma del patrón y calcula la distancia de cada punto. La ventaja de los escáneres 3D de luz estructurada es la velocidad y la precisión. En lugar de escanear un punto a la vez, los escáneres de luz estructurados escanean varios puntos o todo el campo de visión a la vez.

- Técnicas volumétricas [2]: Tenemos la tomografía computarizada que genera una imagen tridimensional del interior de un objeto a partir de imágenes de rayo X bidimensionales, de manera similar tenemos la resonancia magnética. Este método es utilizado en entornos médicos para realizar capturas de tejidos blandos del cuerpo, lo hace útil en imágenes neurológicas, músculo-esqueléticas, cardiovasculares y oncológicas.

4.3 Pasiva sin contacto

Las soluciones de imágenes 3D pasivas no tienen un hardware específico que emiten algún tipo de radiación por sí mismas, sino que se basan en la detección de la radiación ambiental. Los métodos pasivos pueden ser muy económicos, porque en la mayoría de los casos basta con cámaras digitales simples [10]. A continuación se enumerarán algunos métodos de reconstrucción.

- Los sistemas estereoscópicos generalmente emplean dos cámaras de video, ligeramente separadas, mirando la misma escena. Analizando las leves diferencias entre las imágenes vistas por cada cámara, es posible determinar la distancia en cada punto de las imágenes. Este método se basa en los mismos principios que rigen la visión estereoscópica humana[20].

- Los sistemas fotométricos generalmente usan una sola cámara, pero toman múltiples imágenes bajo diferentes condiciones de iluminación. Estas técnicas intentan invertir el modelo de formación de imágenes para recuperar la orientación de la superficie en cada píxel.

5 Conceptos Básicos

5.1 Teoría método clásico

5.1.1 Modelo de cámara

Un modelo de cámara [12] describe la relación matemática entre las coordenadas de un punto en el espacio 3D y su proyección en un plano 2D. Se plantea, entonces, dos aspectos importantes. Por un lado, se tiene una cámara, que se comporta como un sensor que capta los rayos de luz de su entorno. Por tanto, se produce una correspondencia entre rayos y coordenadas de los puntos imagen, definidas según una geometría intrínseca a la propia cámara y sus lentes. Por otro lado, existe una relación entre el marco de referencia de la cámara y el sistema de referencia objeto, son los llamados parámetro externos.

En general, puede afirmarse que la relación existente entre cualquier punto p del espacio y su proyección $p^{'}$ en el plano de la imagen se puede simplificar con la siguiente ecuación [11]:

$$p^{'} = K[R|t]p$$

Donde $p^{'}$ son las coordenadas en píxeles de la proyección, K es una matriz 3x3 en la que se representan los parámetros intrínsecos de la cámara (la distancia focal de la lente de la cámara, el grado de perpendicularidad de las paredes de los píxeles del sensor y el desplazamiento del centro de la imagen). R y t son los parámetros extrínsecos de la cámara. Siendo R una matriz de rotación 3x3 y t es un vector de traslación de la cámara con sistema de referencia en el mundo. p son las coordenadas X, Y y Z del espacio tridimensional.

5.1.2 Calibración

La calibración de una cámara consiste en calcular los parámetros intrínsecos y extrínsecos.

Existen diferentes métodos de calibración de cámaras, entre los más conocidos se encuentran el de Tsai y Zhang [18].

El método de Tsai tiene dos fases, en el primer paso se realiza la conversión de píxeles a milímetros en función de los valores de la cámara y situando el centro del eje óptico en el centro de la imagen, se consigue calcular la orientación del patrón, la traslación, y el factor de proporción. En el segundo paso se utiliza un método de optimización iterativa para calcular la distancia focal, el coeficiente de distorsión y la traslación en Z.

El método de Zhang propone una técnica de calibración que se basa en la observación de un patrón de referencia del que se toman varias imágenes desde diferentes posiciones. El proceso de calibración se realiza en tres pasos, primero se transforma el sistema de coordenadas del mundo en el de la cámara (matriz extrínseca), después se realiza una corrección de la distorsión y luego se obtienen las coordenadas 2D de la imagen (matriz intrínseca).

5.1.3 Disparidad

Basándonos en la visión humana, se conoce como disparidad a la ligera diferencia entre los dos puntos de vista proporcionados por ambos ojos. La disparidad es la forma de percibir profundidad y relieve más utilizada por el cerebro humano, convirtiéndose en la base para la creación de imágenes 3D. El cerebro coge estos dos puntos de vista distintos y los integra, creando así un objeto en tres dimensiones.

Podemos observar en el apartado 'a' de la Fig.1, 2 cámaras y 2 objetos, una 'persona' y una 'palmera'. En el apartado 'b' podemos observar como si superponemos las imágenes capturadas por las cámaras, entre los elementos existe una distancia, que es la disparidad. En el caso de que acercáramos la 'persona' y la 'palmera', esta disparidad aumentaría y en el caso contrario, la disparidad disminuiría. [20].

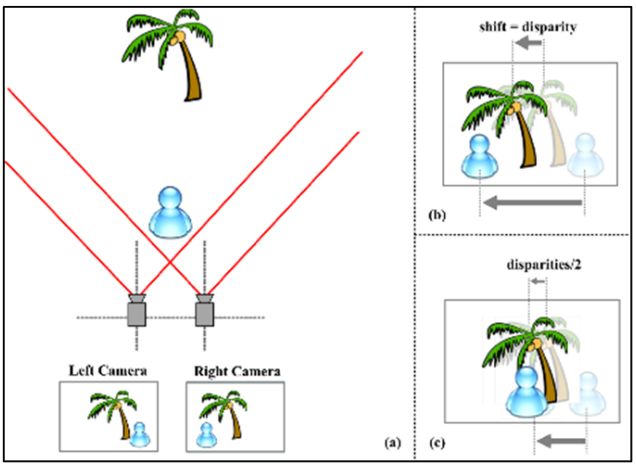


Fig. 1: Visión estereoscópica.

5.1.4 Geometría epipolar

Es muy difícil, en la realidad, conseguir una configuración ideal como la expuesta anteriormente en la que los ejes de las dos cámaras estén perfectamente alineados en paralelo y la línea base sea perpendicular a los dos ejes ópticos. Por ello, es necesario recurrir a la geometría epipolar [3] para lograr obtener la profundidad de los objetos en la imagen. La geometría epipolar se describe de la siguiente manera.

- Si observamos la escena de la Fig. 2, tenemos un punto X que deseamos captar y dos cámaras C_1 y C_2. La proyección 2D resultante de X es el punto X_1 y X_2.

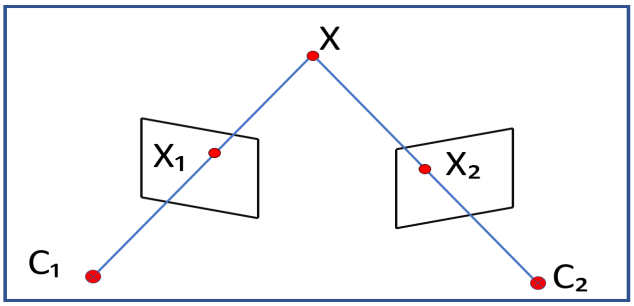


Fig. 2: Escena con 2 cámaras y un punto X.

- Como podemos observar en la Fig. 3 tenemos una línea que une la cámara C_2 con el punto X. Si a partir de aquí empezamos a trazar líneas que se unan con la cámara C_1, hay plano que converge en la proyección 2D, que se denomina línea epipolar e_1.

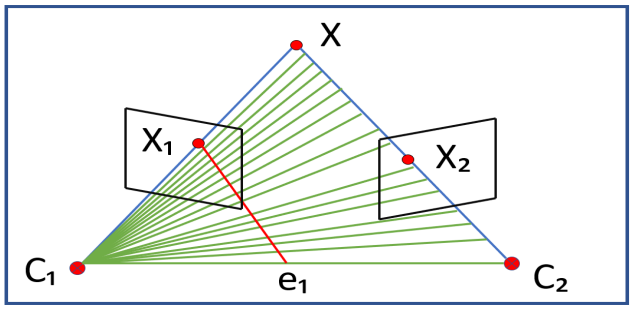


Fig. 3: Trazado de líneas.

- En la Fig. 4 obtendremos un el plano apiolar y la recta inferior denominada epipolo E.

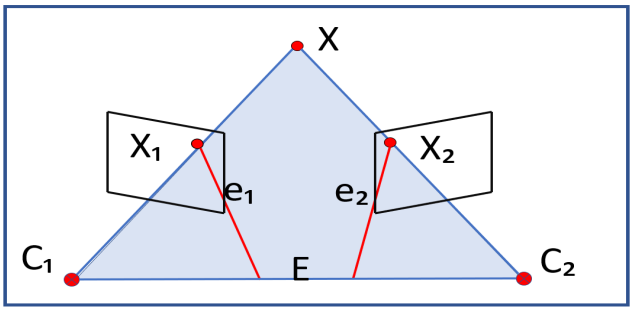


Fig. 4: Plano epipolar.

5.2 Teoría método con IA

5.2.1 Ray tracing

El *ray tracing* [13] es una técnica de renderizado que se utiliza para el cálculo de la intensidad de un pixel y de esta manera realizar efectos globales de iluminación como pueden ser reflexiones, refracciones o sombras.

En el *ray tracing* [13] se parte de la cámara principal, desde esta se emite un rayo cuyo objetivo es impactar contra la primera superficie visible, una vez que tenemos este primer impacto se busca las fuentes de luz, esta fuente de luz puede ser directa o indirecta.

- Directa: que viene de fuentes de luz como el sol, focos y otros y luego la iluminación.
- Indirecta: que viene de objetos que reflejan la luz, que pueden ser espejos, superficies mate brillante y prácticamente cualquier cosa que ilumine las zonas que les rodean.

5.2.2 Renderización volumétrica

Esta técnica permite interpretar los datos tridimensionales que de una escena a partir del uso de rayos. Parecido al *ray tracing*, generamos rayo que parten de la cámara hacia los elementos de la escena, pero en vez de rebotar en el objeto y generar rayos secundarios, lo que sé hacer es atravesar el objeto. En este tipo de técnicas se trabaja con data sets tridimensionales volumétricos en los que contamos con puntos de información distribuidos por todo el espacio interior del objeto. NeRF está basado en el funcionamiento de la renderización volumétrica [14], más específicamente en la técnica de *volume ray casting*, a veces denominado *volumetric ray casting, volumetric ray tracing o volume ray marching*. Como se puede observar en la Fig. 5 este algoritmo consta de cuatro pasos:

1. Lanzamiento de rayos: Para cada píxel de la imagen, se proyecta un rayo a través del volumen.
2. Muestreo: por el recorrido del rayo se seleccionan puntos de muestreo aleatorios. Pero por lo general estos puntos se encuentran entre los voxels, es necesario interpolar los valores de los puntos de muestreo con los voxels vecinos (este algoritmo trabaja con imágenes volumétricas, y la unidad mínima procesable se denomina volxel, equivalen al pixel en una imagen 2D).
3. Sombreado: para cada punto de muestreo. Se utiliza una función de transferencia. Se coge información del *data set* como *input* y devuelve como resultado un color *RGB* y una densidad α.
4. Composición: una vez que se han sombreado todos los puntos de muestreo, se componen a lo largo del rayo de vista, lo que da como resultado el valor de color final para el píxel que se está procesando actualmente. Es similar a mezclar láminas de acetato en un retroproyector.
5. Se repite este proceso para cada píxel en la imagen. Y de esta manera se obtiene una escena renderizada por el método de *volume ray casting*.

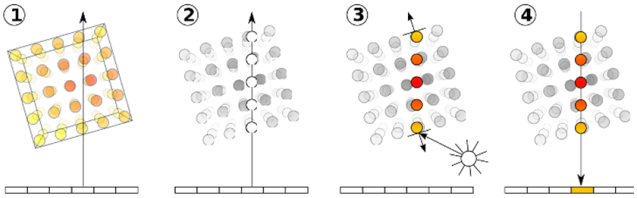


Fig. 5: Pipeline *volume ray casting*

5.2.3 Funcionamiento NeRF

El funcionamiento del algoritmo es el siguiente:

- Primero se toman diferentes imágenes registrando diferentes perspectivas.
- A continuación, NeRF utiliza el algoritmo COLMAP[16] (es una línea de reconstrucción 3D basada en SfM[22] y MVS[17] *Multi-View Stereo*) para establecer las coordenadas relativas del conjunto de imágenes. De esta manera conocer la dirección y la posición relativa desde donde se han realizado las imágenes.
- Como en el apartado anterior, se lanza unos rayos desde la cámara a través de la escena para generar un conjunto de puntos de muestreo.
- Pero ahora tenemos la diferencia, que estamos trabajando con imagenes 2D y no con un *data set* volumétrico. Por lo tanto, los puntos de muestreo no tienen una referencia para generar un color.
- Aquí viene el punto innovador del modelo NeRF, este utiliza una red neuronal para generar los colores en estos puntos de muestreo. En la primera iteración estos colores serán aleatorios, pero al tener una imagen de referencia podemos observar el error producido y mediante el descenso de gradiente podemos ir minimizando este error.
- Teniendo en cuenta que hemos tomado fotografías desde diferentes ángulos de vista, estamos minimizando el error de estas múltiples vistas simultáneamente. Como podemos observar en la Fig. 6 al conocer la posición y dirección de las imágenes y podemos realizar un proceso de triangulación para ir produciendo un objeto tridimensional.

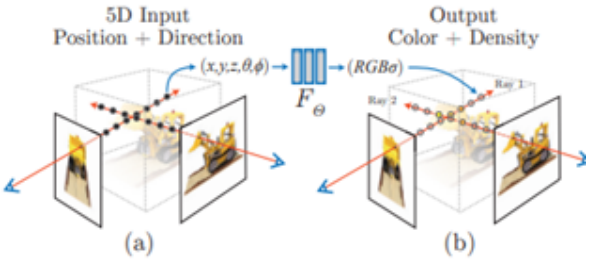


Fig. 6: Funcionamiento de NeRF.

5.3 Implementaciones

5.3.1 Calibración

A continuación, se explicará una implementación de una calibración realizada con la librería OpenCV, que hace uso del método Zhang. La implementación se realizó siguiendo los siguientes pasos.

- Primero hay que realizar varias capturas a un elemento, que contenga un patrón de referencia. En nuestro caso se ha escogido un tablero de ajedrez.
- A continuación, hay que encontrar las esquinas internas de nuestro tablero para emplearlo como marcadores de calibración. Esto se puede conseguir empleando la función *findChess-boardConers()*[7] de la librería OpenCV, este método hace uso del cambio de intensidades y de la geometría del tablero para identificar las esquinas internas. El resultado obtenido lo podemos observar en la Fig. 7.

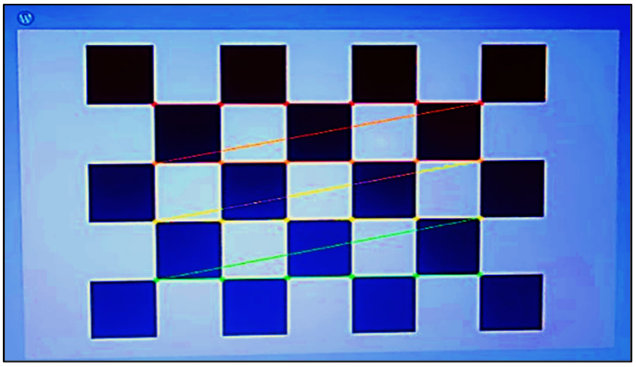


Fig. 7: Esquinas encontradas en tablero de ajedrez.

- Una vez se obtienen los puntos, se utiliza la función *calibrateCamera()*[7], esta función hace uso el método Zhang[18], que aprovecha la separación equidistante de las equinas, que matemáticamente se pueden representar como un patrón ortogonal y basándose en un procedimiento de extracción de fase espacial poder calcular los parámetros de la cámara.
- Como podemos observar en la Fig. 8 una vez calculado los parámetros de la cámara, estos se pueden utilizar para realizar una homografía para proyectar una figura 3D en nuestra imagen. Esta técnica la podemos encontrar en aplicaciones como la realidad aumentada.

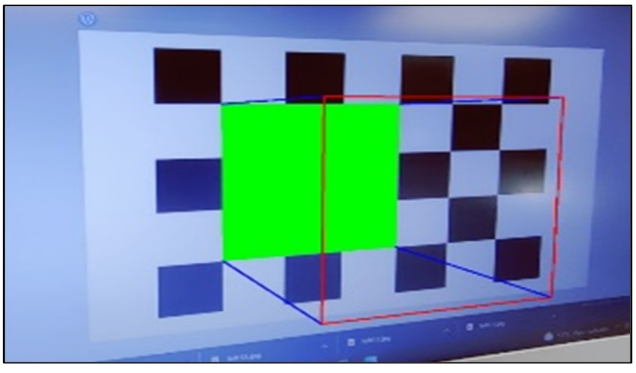


Fig. 8: Proyección de una figura 3D.

5.3.2 Mapa de disparidad con imágenes alineadas

Para este estudio se ha hecho uso del dispositivo Intel Realsense D435, este dispositivo consta de dos cámaras alineadas. Esta característica nos permite simplificar el proceso de obtención de información tridimensional. Como podemos observar en la Fig. 9 al tener las dos imágenes alineadas una de la otra, simplemente con ayuda de un método de comparación, podremos encontrar la disparidad que hay entre una imagen y la otra. De esta manera poder generar el mapa de profundidad.

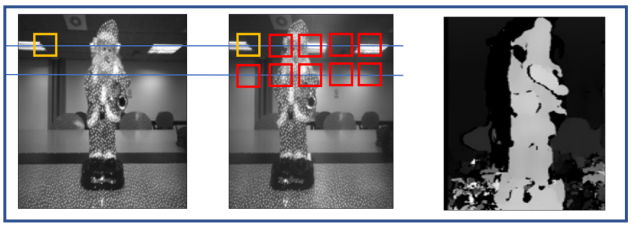


Fig. 9: Mapa de profundidad con imágenes alineadas.

5.3.3 Mapa de disparidad con imágenes no alineadas

En este apartado estudiaremos un método pasivo sin contacto, hemos visto que podemos capturar la escena de barias maneras, podemos tener dos cámaras o una sola cámara, pero tomar múltiples imágenes.
En esta apartado se explicará la implementación con una sola cámara y un par de imágenes. Para realizar el desarrollo se ha hecho uso de la librería OpenCV y se han seguido los siguientes pasos:

- Primero, se ha hecho 2 capturas, a mano alzada de un objeto, estas imágenes están ligeramente desplazadas una de la otra.
- El siguiente paso es emplear un algoritmo SIFT*(Scale Invariant Feature Transform)* [19]. Este algoritmo escanea la imagen utilizando ventanas de diferentes tamaños y haciendo uso de la diferencia gaussiana, como podemos ver en la Fig. 10 gracias a este método se puede extraer *keypoints* (puntos de interés).

Fig. 10: Resultado de aplicar algoritmo SIFT.

- En el apartado anterior se calculó una serie de *keypoints*, pero hay diferentes para cada imagen. Para este apartado se filtran los puntos que coinciden en ambas imágenes, para esta tarea se utiliza el algoritmo FLANN *(Fast Library for Approximate Nearest Neighbors)*[9]. Primero se coge un punto de referencia en la

dos imágenes, y se aceptarán los *keypoints* como coincidentes, si la distancia a estos puntos de referencia están por debajo de un umbral. En la Fig. 11 se puede observar algunos puntos coincidentes.

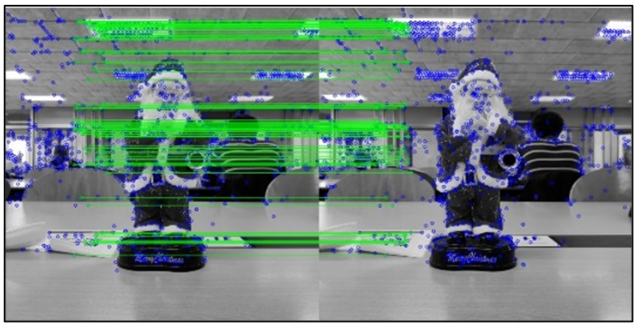


Fig. 11: Resultado de aplicar algoritmo FLANN.

- El siguiente paso es aplicar la geometría epipolar [3]. Como podemos observar en la Fig. 12 las líneas epipolares resultantes tienen un cierta inclinación, al realizar las capturas de las imágenes a manos alzada, estas no están perfectamente alineadas, para ello hay que rectificar las imágenes, y proyectarlas en un nuevo plano común.

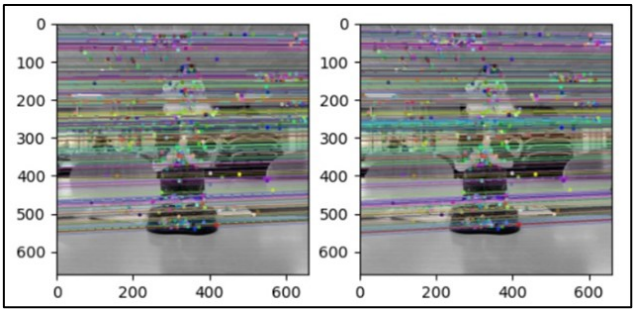


Fig. 12: resultado de aplicar la geometría epipolar.

- Una vez tenemos las imágenes rectificadas, es cuestión de emplear un algoritmo de comparación, para encontrar la disparidad que hay entre una imagen y la otra. Como podemos observar en la Fig. 13 obtendremos el mapa de profundidad.

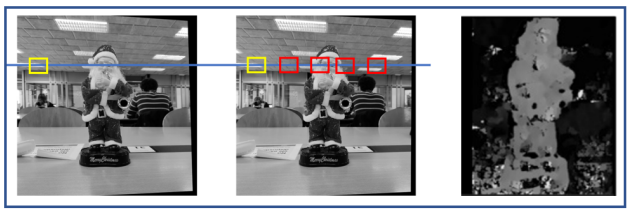


Fig. 13: Mapa de profundidad con imágenes no alineadas.

6 Experimento y Resultados

6.1 Reconstrucción 3D con cámara estereoscópica

En este apartado se realizará una implementación activa sin contacto con el dispositivo *Intel Realsense D435*. Este dispositivo consta de una cámara estereoscópica (dos cámaras alineadas con una separación entre ellas) y un proyector de infrarrojos que permite sondear objetos mediante la técnica de tiempo de vuelo vista anteriormente. Para hacer uso del dispositivo y realizar las capturas, disponemos de un software dedicado nombrado *Intel RealSense SDK 2.0*, el inconveniente de este programa es que no permite realizar el escaneo completo del objeto. Por lo que se tenía que ir tomando capturas de diferentes partes del objeto y posteriormente unirlas, para esta tara se ha utilizado el programa *CloudCompare*[1].

A continuación se detalla el procedimiento.

- Como el dispositivo genera una figura con baja resolución, el primer paso era encontrar un objeto adecuado para la reconstrucción. Como se puede observar en la Fig. 14 se probaron diferentes objetos.

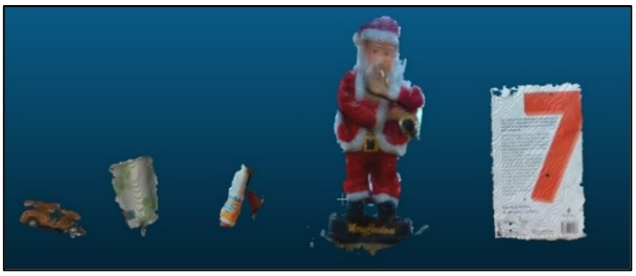


Fig. 14: Escaneo de diferentes objetos, con *Intel Realsense D435*.

- El dispositivo capturaba toda la escena, por lo tanto, el primer paso era recortar el sobrante.
- En la Fig. 15 se puede ver la herramienta del *CloudCompare*, el cual permite indicar puntos en común entre las dos capturas para realizar una unión.

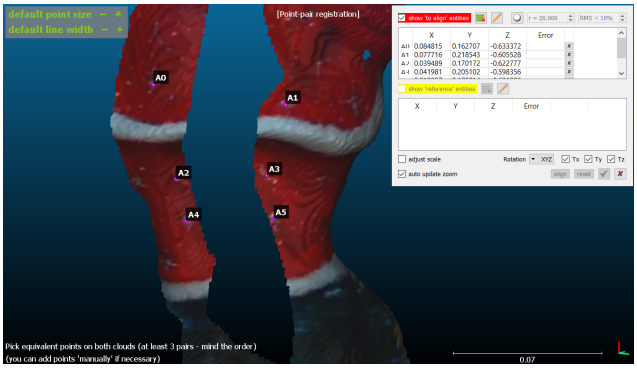


Fig. 15: Herramienta *merge* de *CloudCompare*

- Después de cada unión se ha seguido recortando las partes que no acabaran de encajar. Se ha repetido este procedimiento con todas las partes hasta obtener la figura completa.

- En la Fig. 16 se puede observar el conjunto de capturas finales.

Fig. 16: Recortes finales.

- En la Fig. 18 y el Fig. 18 se puede apreciar el resultado final de la reconstrucción.

Fig. 17: Frontal reconstrucción 3D.

Fig. 18: Dorsal reconstrucción 3D.

- Este procedimiento consume mucho tiempo y esfuerzo para ser realizarlo.

6.2 Reconstrucción 3D con método clásico

En este apartado se realizará una implementación pasiva sin contacto con el algoritmo SfM *(Structure from Motion)* [22]. SfM es una técnica clásica que unifica los conceptos vistos hasta el momento (identificación de características, correspondencia de puntos, parámetros de la cámara, geometría epipolar, etc), método fácil de utilizar, capaz de representar un objeto 2D a 3D con solo unas cuantas fotografías desde diferentes puntos de vista Los pasos que sigue el algortimo son los siguientes:

- En primer lugar, hay que tomar diferentes imágenes del objeto que queremos reconstruir desde diferentes puntos de vista.
- A continuación, se realiza una identificación de características comunes en las diferentes imágenes que permita relacionarlas. Un popular acercamiento a este problema es el sistema de extracción de características SIFT. Estas características invariantes reciben el nombre de *keyponints* (puntos de interes).
- Después, se realiza una correspondencia de estos *keyponints*.
- Posteriormente, se extraen los parámetros de la cámara.
- Por último, se hace uso de la geometría epipolar para ubicar los puntos en común en cada fotografía, primero se toma en cuenta la posición de la cámara, la orientación y la geometría de la escena y se realizan las proyecciones mediante operaciones matemáticas, es así como se van generando las nubes de puntos.

Para realizar la reconstrucción se ha hecho uso del software *Meshroom* que automatiza todo este proceso. Para producir un resultado es cuestión de realizar las fotografías y esperar a que el software genere la reconstrucción.

- Primero se tomó 40 fotografías del objeto a reconstruir.
- En la Fig. 19 se puede ver la interfaz del software.

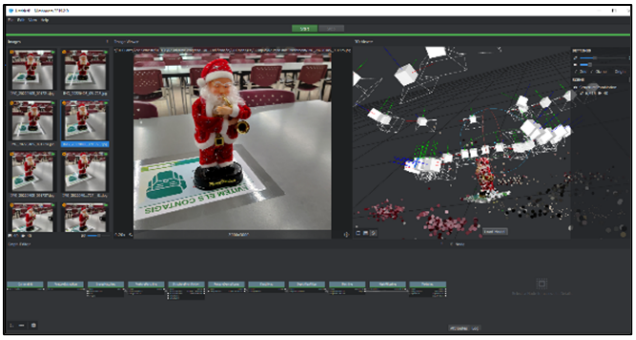


Fig. 19: Interfaz *MeshRoom*

- En la Fig. 20 y en la Fig. 20 podemos ver el resultado obtenido.

Fig. 20: Frontal reconstrucción 3D.

Fig. 21: Dorsal reconstrucció 3D.

6.3 Reconstrucción 3D con IA

En este apartado se realizará una implementación pasiva sin contacto con el modelo de aprendizaje computacional NeRF (*Neural Radiance Fields*)[21][4] [22]. NeRF es una nueva técnica de renderización que desde el 2020 compite con las técnicas clásicas de fotogrametría, que haciendo uso de varias imágenes en dos dimensiones es capaz de reconstruir una escena tridimensional hiperrealista de un lugar u objeto. El algoritmo representa una escena utilizando

una red profunda (no convolucional) completamente conectada, cuya entrada es una única coordenada 5D continua que consta de unaubicación espacial (X, Y, Z) y dirección de visualización (θ , ϕ). Cuya salida es un color *RGB* y una densidad α. El sistema NeRF se estaría basando en lo que se conoce como *volume rendering*[14] (renderización volumétrica).

6.3.1 Resultados NeRF

Para realizar la prueba se ha hecho uso de las mismas imágenes que en el caso de la reconstrucción clásica (40 fotografías. Para poner en marcha el modelo primero hay que ejecutar el algoritmo COLMAP[16], este proceso llevó 4'35", el cual genera un JSON necesario por el modelo, es de remarcar que una vez generado el fichero el modelo de NeRF[21] es capaz de realizar la reconstrucción en unos segundos, en este caso en 38 segundo.

En la Fig. 22 y el Fig. 23 podemos ver los resultados obtenidos por el modelo.

Fig. 22: Frontal reconstrucción 3D.

Fig. 23: Dorsal reconstrucció 3D.

7 Comparativa

En este apartado se realizará una comparativa entre los métodos de reconstrucción por método clásico y el que utiliza aprendizaje computacional. No se ha incluido la reconstrucción con cámara estereoscópica, ya que este método consume mucho tiempo.

En el anexo podemos encontrar todas las pruebas realizadas. En la Fig. 24 y en la Fig. 25 podemos observar como los dos métodos consiguen resultados muy similares. Pero si observamos la Fig. 26 y la Fig. 27 podemos observar como el método SfM es incapaz de realizar la reconstrucción cuando el objeto es transparente, esto también se puede ver en la Fig.30 y la Fig.31. Podemos concluir que los dos métodos obtendrán resultados similares siempre y cuando el objeto no tenga reflejos, transparencias y tenga una textura reconocible. En los otros casos podemos observar una clara superioridad con el método NeRF.

8 Conclusiones

Respecto con los objetivos del proyecto, se han logrado satisfactoriamente. Después de realizar el estudio y analizar los resultados se han llegado a las siguientes conclusiones:

Fig. 24: Reconstrucción 3D NeRF (45 imágenes input).

Fig. 25: Reconstrucción 3D SfM (45 imágenes input).

Fig. 26: Reconstrucción 3D NeRF (54 imágenes input).

Fig. 27: Reconstrucción 3D SfM (54 imágenes input).

Fig. 28: Reconstrucción 3D NeRF (60 imágenes input).

Fig. 29: Reconstrucción 3D SfM (60 imágenes input).

Fig. 30: Reconstrucción 3D NeRF (31 imágenes input).

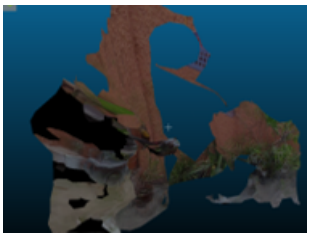


Fig. 31: Reconstrucción 3D SfM (31 imágenes input).

- Si se requiere de una reconstrucción altamente precisa, el mejor método es utilizar un escáner de contacto como la máquina MMC, pero la desventaja de estas es el precio, el cual ronda entre los 50.000€ y 200.000 €.
- Tenemos escáneres activos sin contacto como los ultrasonidos y rayos X o por resonancia magnética, que son muy útiles para la medicina. Y por ahora no son sustituibles.
- En este estudio se ha visto otra implementación activa sin contacto, con la cámara Intel Realsense D435, que a pesar de no obtener un resultado de mucha calidad, hay otros escáneres con las mismas bases, con muy

buenos resultados y muy precisos. El inconveniente es que necesitan un hardware particular que en muchos casos puede llegar a ser muy costos, con precios que rondad entre 400 € a 10000 €.

- Por último hemos visto técnicas de fotogrametría que empleaban SfM o modelos como NeRF, en este caso vemos un gran potencial en el modelo de aprendizaje NeRF y como en algunos casos supera con creces al método clásico. Es muy posible que uno de los mayores beneficiado sea la industria del entretenimiento y que el uso de esta técnica reduzca el proceso de modelado.

9 FUTURAS IMPLEMENTACIONES

Resultaría interesante utilizar el modelo NeRF para realizar diferentes implementaciones como:

- Generar modelos para realizar videojuegos.
- Reconstruir el interior de una casa y combinarlo con VR, para realizar visitas remotas.
- Reconstruir productos y realizar una tienda online.

REFERÈNCIES

[1] Cloud Compare Descarga. https://www.danielgm.net/cc/, 2022.

[2] Escaneo 3D. https://hmong.es/wiki/3D_scanning, 2022.

[3] Geometría Epipolar. https://hmong.es/wiki/Epipolar_geometry, 2022.

[4] Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. https://nvlabs.github.io/instant-ngp/, 2022.

[5] Intel Realsense. https://www.intel.es/content/www/es/es/products/sku/128255/intel-realsense-depth-camera-d435/specifications.html, 2022.

[6] Meshroom. https://alicevision.org/#meshroom, 2022.

[7] OpenCV: Camera Calibration. https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html, 2022.

[8] OpenCV: Camera Calibration and 3D Reconstruction. https://docs.opencv.org/4.x/d9/db7/tutorial_py_table_of_contents_calib3d.html, 2022.

[9] OpenCV: Feature Matching with FLANN. https://docs.opencv.org/4.x/d5/d6f/tutorial_feature_flann_matcher.html, 2022.

[10] Reconstrucción 3D. https://hmong.es/wiki/3D_reconstruction, 2022.

[11] Julián Aguirre de Mata. *Calibración geométrica de cámaras no métricas. Estudio de metodologías y modelos matemáticos de distorsión*. PhD thesis, Topografia, 2016.

[12] José Javier Alcalde Sanz. Diseño de un protocolo de calibración de cámaras estéreo. B.S. thesis, 2014.

[13] Andrés Navarro Cadavid, Dinael Guevara Ibarra, and María Victoria Africano. Calibración basada en medidas para modelos de trazado de rayos en 3d para ambientes exteriores urbanos andinos. *Sistemas & Telemática*, 10(21):43–63, 2012.

[14] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4):65–74, 1988.

[15] Julien Faucher, Manuel BOISSENIN, Bala AMAVASAI, and Jon R TRAVIS. Camera calibration and 3-d reconstruction. Technical report, Technical report, June 2006. 141, 2006.

[16] Alex Fisher, Ricardo Cannizzaro, Madeleine Cochrane, Chatura Nagahawatte, and Jennifer L Palmer. Colmap: A memory-efficient occupancy grid mapping framework. *Robotics and Autonomous Systems*, 142:103755, 2021.

[17] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

[18] José Isern González. *Estudio experimental de métodos de calibración y autocalibración de cámaras*. PhD thesis, 2003.

[19] David G Lowe. Method and apparatus for identifying scale invariant features in an image and use of same for locating an object in an image, March 23 2004. US Patent 6,711,293.

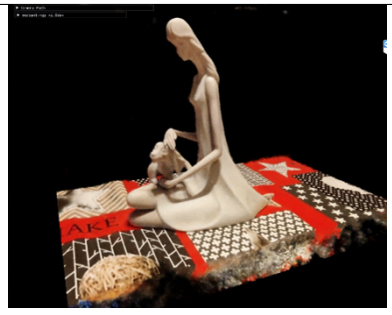
[20] Santiago Martín, Javier Suárez, Ramón Rubio, and Ramón Gallego. Aplicación de los sistemas de visión estereoscópica en las enseñanzas técnicas. *Escuela de Ingenieros Técnicos Industriales de Gijón. Universidad de Oviedo*, 2004.

[21] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

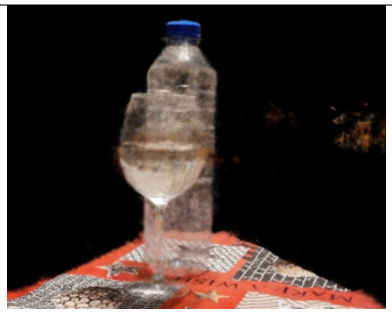
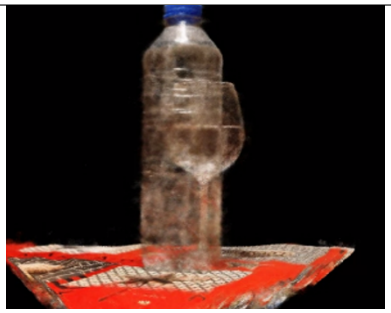
[22] Roberto Tomás-Jover, Adrián J Riquelme Guill, Miguel Cano González, Antonio Abellán Fernández, and Luis Jordá. Structure from motion (sfm): una técnica fotogramétrica de bajo coste para la caracterización y monitoreo de macizos rocosos. In *Reconocimiento, tratamiento y mejora del terreno: 10º simposio nacional de ingeniería geotécnica: A coruña, 19, 20 y 21 de octubre de 2016*, pages 209–216. Sociedad Española de Mecánica del Suelo e Ingeniería Geotécnica, 2016.

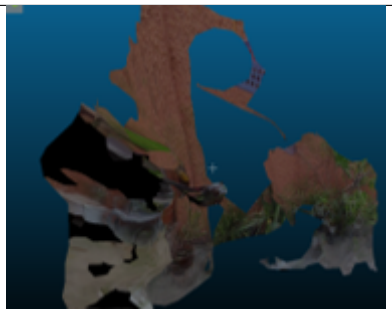
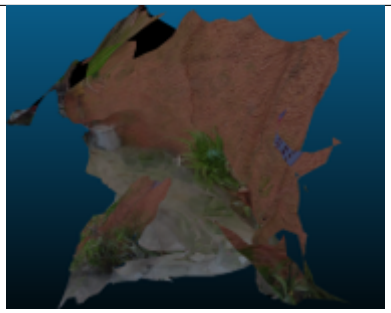
.

Anexo

Input (40 imgs)			
SfM 4’45”			
NeRF 5’13” COLMAP 4’35” Modelo 38”			
Input (45 imgs)			
SfM 5’08”			
NeRF 5’57” COLMAP 5’12” Modelo 45”			

. Anexo

Input (54 imgs)			
SfM 6’35”			
NeRF 6’25” COLMAP 5’45” Modelo 40”			
Input (60 imgs)			
SfM 7’24”			
NeRF 8’06” COLMAP 7’16” Modelo 50”			

Input (31 imgs)			
SfM 7’05”			
NeRF 6’19” COLMAP 5’49” Modelo 30”