

3D-PROJECT

INGENIERÍA DE MANTENIMIENTO DE COMPUTADORES Y REDES

Grupo 1

Vadym Formanyuk

Daniel Asensi Roch

Álvaro Martínez García

Alfonso Izura Concellón

Juan Llinares Mauri

Edgar Verdu Sanchez

Grupo 2

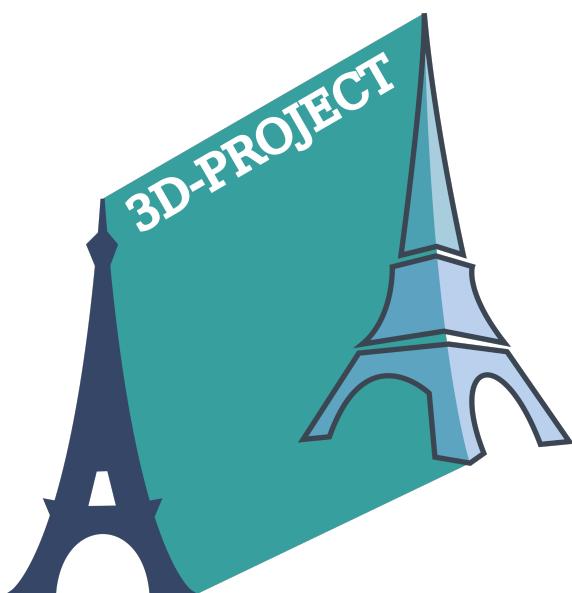
Alexander Andonov Aracil

David Herrero Arráez

Carmen Hidalgo Brotons

Alejandro Torres Montero

Jorge Mirete Hernández



Última revisión: 25 de noviembre de 2023

Índice

PARTE: 1 Recuperación de imágenes 2D	3
1. Captación de Imágenes 2D mediante el uso de BigData	3
1.1. Qué es la captación	3
1.2. Por qué el BigData	3
1.3. Como se realiza	4
1.4. Estado del arte	4
2. Preprocesado de Imágenes 2D	6
2.1. Qué es el preprocesado	6
2.2. Por que hay que realizar el preprocesado	6
2.3. Cómo se realiza	7
2.4. Estado del arte	7
3. Indexado de Imágenes 2D	9
3.1. Qué es el indexado	9
3.2. Por que hay que realizar la indexación	9
3.3. Como se realiza	9
3.4. Estado del arte	10
PARTE: 2 Reconstrucción 3D	12
4. Introducción	12
4.1. Motivación	12
4.2. Contexto	12
4.3. State of Art	13
4.3.1. Extracción de características	13
4.3.2. Matching: Cálculo de Correspondencias	14
4.3.3. Camera Pose Estimation	15
4.3.4. Reconstrucción Estéreo	16
4.3.5. Visualización	17
5. Marco Teórico	18
5.1. Pipeline	18
5.2. Descriptores y correspondencias	18
5.3. Bundle Adjustment	19
5.3.1. ¿Qué es?	19
5.3.2. ¿Cómo se obtiene?	19
5.3.3. ¿Qué obtenemos?	20
5.4. Multiview Stereopsis (MVS)	20
5.5. Nube de puntos	21
5.5.1. ¿Qué es?	21
5.5.2. ¿Cómo se obtiene?	22
5.5.3. ¿Qué se puede generar?	22
5.6. Visualización del modelo de malla 3D	23
PARTE: 3 Plan de Financiación	24

6. ...	24
PARTE: 4 Plan de Mantenimiento	25
7. ...	25

Recuperación de imágenes 2D

La captación de imágenes 2D mediante el uso de Big Data implica el uso de herramientas y técnicas de procesamiento de imágenes y aprendizaje automático para analizar grandes conjuntos de imágenes y extraer información útil. Las ventajas incluyen escalabilidad, velocidad, precisión y personalización. El preprocesado de imágenes 2D es una parte fundamental del proceso de creación de modelos 3D, y se realiza mediante la mejora de la calidad de las imágenes, la segmentación de la imagen, la extracción de características y la normalización de las imágenes. El preprocesado es crucial para la creación de modelos 3D precisos y útiles. El estado del arte en ambos campos está en constante evolución debido al rápido desarrollo de nuevas técnicas y tecnologías en el campo del aprendizaje automático, la inteligencia artificial y el análisis de datos.

1 Captación de Imágenes 2D mediante el uso de BigData

1.1. Qué es la captación

La captación de imágenes 2D mediante el uso del BigData se refiere a la gestión y análisis de grandes conjuntos de datos, que pueden incluir datos estructurados (como bases de datos) y datos no estructurados (como imágenes, videos y texto). La idea es tener la capacidad de procesar grandes cantidades de imágenes de forma rápida y eficiente.

El uso de Big Data en la captación de imágenes 2D implica el uso de herramientas y técnicas de procesamiento de imágenes y aprendizaje automático para analizar grandes conjuntos de imágenes y extraer información útil. Por ejemplo, se pueden utilizar algoritmos de aprendizaje automático para identificar patrones en las imágenes, clasificar objetos en las imágenes y detectar objetos en movimiento.

1.2. Por qué el BigData

Hay varias ventajas de la utilización del Big Data en la captación de imágenes 2D en comparación con otros procesos:

- 1) **Escalabilidad:** El procesamiento de grandes cantidades de datos es uno de los principales beneficios del Big Data. La capacidad de procesar grandes conjuntos de imágenes de forma eficiente permite a los usuarios realizar análisis más detallados y precisos.
- 2) **Velocidad:** Los algoritmos de Big Data están diseñados para procesar grandes cantidades de datos en tiempo real, lo que significa que los usuarios pueden obtener resultados rápidamente. Esto es especialmente importante en situaciones en las que se necesita tomar decisiones rápidas basadas en la información de las imágenes.
- 3) **Precisión:** Los algoritmos de Big Data pueden analizar grandes conjuntos de imágenes para identificar patrones y tendencias que pueden ser difíciles de

detectar a simple vista. Esto puede ayudar a los usuarios a obtener información más precisa y completa sobre los datos.

- 4) **Personalización:** La utilización de Big Data permite a los usuarios personalizar sus análisis en función de sus necesidades específicas. Los usuarios pueden elegir los algoritmos y herramientas de análisis que mejor se adapten a sus necesidades y objetivos.

1.3. Como se realiza

El proceso de captación de imágenes 2D mediante el uso de Big Data puede involucrar varias etapas, como se describe a continuación:

- 1) **Recopilación de datos:** La primera etapa es la recopilación de imágenes en 2D de diversas fuentes, como cámaras de vigilancia, satélites, redes sociales, entre otras. Estas imágenes son almacenadas en un repositorio centralizado.
- 2) **Preprocesamiento de datos:** El siguiente paso es el preprocesamiento de los datos, que puede involucrar la limpieza y normalización de los datos, así como la eliminación de imágenes redundantes o irrelevantes.
- 3) **Extracción de características:** En esta etapa, se extraen características de las imágenes mediante técnicas de procesamiento de imágenes, como la extracción de bordes, la segmentación y la identificación de patrones. Estas características se utilizan como entradas para el análisis posterior.
- 4) **Análisis de datos:** El análisis de datos implica el uso de técnicas de Big Data, como el aprendizaje automático, la minería de datos y la visualización de datos, para identificar patrones y tendencias en los datos de las imágenes.
- 5) **Aplicación de resultados:** Finalmente, los resultados del análisis se aplican a una variedad de aplicaciones, como el reconocimiento de objetos, la detección de anomalías, la clasificación de imágenes y el seguimiento de objetos.

En resumen, el proceso de captación de imágenes 2D mediante el uso de Big Data involucra la recopilación de imágenes de diversas fuentes, el preprocesamiento de los datos, la extracción de características, el análisis de datos y la aplicación de resultados para una variedad de aplicaciones.

1.4. Estado del arte

El estado del arte en captación de imágenes mediante el uso de Big Data está en constante evolución debido al rápido desarrollo de nuevas técnicas y tecnologías en el campo del aprendizaje automático, la inteligencia artificial y el análisis de datos.

Algunas de las tendencias actuales en este campo incluyen:

- **Aprendizaje profundo:** El uso de técnicas de aprendizaje profundo, como redes neuronales convolucionales (CNN), para la extracción de características de imágenes es una tendencia creciente. Estas técnicas han demostrado tener un rendimiento sobresaliente en la clasificación y detección de objetos en grandes conjuntos de datos de imágenes. <https://www.sciencedirect.com/science/article/pii/S1051200422001312>

- **Procesamiento de imágenes en tiempo real:** Con el aumento de la velocidad de procesamiento de los sistemas de cómputo, se está trabajando en técnicas que permitan el procesamiento en tiempo real de grandes cantidades de datos de imágenes. Esto permitiría la detección y clasificación de objetos en tiempo real, lo que tendría aplicaciones en seguridad y vigilancia. <https://ieeexplore.ieee.org/document/7920232>
- **Integración con otros datos:** La captación de imágenes 2D también se está integrando con otros tipos de datos, como datos de sensores y datos de redes sociales, para crear conjuntos de datos más amplios y completos que puedan proporcionar información más útil y precisa. <https://www.sciencedirect.com/science/article/pii/S016786551830045X>
- **Análisis de video:** Además de la captación de imágenes estáticas, se está trabajando en el análisis de video mediante el uso de Big Data. Esto permite el seguimiento de objetos en movimiento y la detección de eventos en tiempo real. <https://ieeexplore.ieee.org/abstract/document/4906450>

2 Preprocesado de Imágenes 2D

2.1. Qué es el preprocesado

El preprocesado de las imágenes 2D captadas es una parte fundamental del proceso de creación de modelos 3D. Este proceso se encarga de preparar las imágenes para su posterior procesamiento, de manera que se pueda extraer la información necesaria para construir un modelo 3D preciso.

Podemos dividir el preprocesado en una serie de subprocesos que podrían incluir:

- Mejora de la calidad de las imágenes: Se utilizan técnicas de mejora de la calidad de las imágenes para corregir problemas como el ruido, la distorsión o la iluminación. Esto ayuda a mejorar la calidad de los datos y, por lo tanto, a generar modelos 3D más precisos.
- Segmentación de la imagen: La segmentación es el proceso de dividir una imagen en varias partes o regiones para analizarlas por separado. Se utiliza para identificar objetos en la imagen y separarlos de su entorno. Esto puede ayudar a mejorar la calidad de los datos y reducir el ruido.
- Extracción de características: La extracción de características implica identificar características específicas en la imagen, como formas, colores o texturas, que pueden utilizarse para construir un modelo 3D preciso. Esto se puede hacer utilizando técnicas de procesamiento de imágenes como la convolución o la detección de bordes.
- Normalización de las imágenes: La normalización es el proceso de convertir las imágenes a un formato estándar para que puedan ser procesadas por el software. Esto puede incluir la conversión de formatos de archivo o la estandarización de la resolución y el tamaño de la imagen.

2.2. Por qué hay que realizar el preprocesado

La necesidad del preprocesado de imágenes 2D radica en la complejidad de los datos y la dificultad para extraer la información necesaria para la construcción de modelos 3D precisos. Las imágenes captadas pueden ser de baja calidad, tener ruido o estar afectadas por la iluminación o las sombras, lo que puede afectar negativamente la calidad del modelo final.

Además, la creación de modelos 3D a partir de big data requiere la extracción de información de una gran cantidad de imágenes, lo que aumenta la complejidad del proceso. El preprocesado de imágenes ayuda a simplificar y mejorar el proceso al reducir el ruido y mejorar la calidad de los datos. También permite una segmentación más precisa de las imágenes y una extracción más efectiva de características relevantes.

El preprocesado de imágenes es una etapa fundamental del proceso de creación de modelos 3D basados en big data. La calidad de los datos y la precisión del modelo final dependen en gran medida de la calidad del preprocesado. Por lo tanto, la necesidad de un preprocesado efectivo y eficiente es crucial para la creación de modelos 3D precisos y útiles.

2.3. Cómo se realiza

Para realizar este preprocesado de las imágenes para que se puedan tratar en las siguientes etapas, se puede realizar con las siguientes herramientas:

- **Redimensionamiento:** Las imágenes se deben redimensionar para obtener unos tamaños y escalas acordes. De esta forma, podremos facilitar la triangulación de puntos comunes. Esto es más fácil con el uso de imágenes vectoriales, que no empeorarían la calidad de la imagen, aunque se suelen utilizar otros formatos más comunes.
- **Corrección de iluminación:** Las imágenes, según muchos factores distintos como la hora del día o el contraste de la cámara, pueden representar el objeto con iluminaciones muy distintas, por lo que se pueden aplicar algunas técnicas para mejorarlo, como la ecualización de histogramas que representen a la imagen para que redistribuya los valores de intensidad. Esto también se podría aplicar a posibles filtros que tuvieran las imágenes, aunque dependen del tipo de filtro que sea.
- **Filtro de ruido:** Las imágenes pueden presentar ruido que empeoren la calidad de la imagen. Para solucionarlo, se pueden aplicar filtros para eliminar el ruido, siendo los más comunes el filtro de mediana y el filtro gausiano.
- **Detección de características Scale-invariant feature transform (SIFT):** Para triangular los puntos característicos del objeto en fotos diferentes, podemos identificar estas características con técnicas como SIFT. Sin embargo, dado a la naturaleza de nuestras imágenes y nuestro objetivo, la extracción mediante SIFT puede no ser suficiente y la extracción mediante ASIFT podría ofrecernos más y mejor información. Esto se debe a que SIFT simula 3 parámetros, mientras que ASIFT simula 6: zoom, ángulo de la cámara en latitud, ángulo de la cámara en longitud y normalización de los parámetros de traslación, rotación y escala. [31]
- **Normalización:** A partir de las características y la información de la cámara, para tener una escala y orientación comunes, podemos utilizar puntos de referencia para normalizar las imágenes.

2.4. Estado del arte

Con lo que respecta al estado del arte, se han encontrado varios papers que tratan también imágenes en 2 dimensiones, que, aunque no los usen para el mismo objetivo, el preprocesamiento de imágenes sigue siendo necesario en todos ellos. Algunos de ellos son:

- **Procesamiento de imágenes:** Este paper trata sobre el procesamiento de las imágenes. En él, cuando habla de los componentes de un sistema de visión, menciona el preprocesamiento como uno de ellos. Menciona el uso de técnicas para la reducción de ruido y el realce de detalles y destaca el uso de la segmentación como un método de preprocesamiento para extraer objetos segmentando la imagen y poder extraerlos y analizarlos. [13]

- **Preprocesamiento de imágenes digitales a través de su Transformada de Fourier:** Este paper analiza los filtros Herméticos Gaussianos aplicados a la transformada de Fourier. Sienta las bases de lo que es un filtro y cómo podemos preprocesar las imágenes con los filtros convencionales y Gaussianos para extraer su información, viendo cómo se mejora su calidad utilizando la transformada de Fourier [27]
- **Procesamiento Digital de Imágenes:** Este libro explica las bases del procesamiento digital de imágenes, subdividiéndolo en 5 áreas principales y siendo una de ellas el preprocesado. Incluye dentro del preprocesado la reducción de ruido y el realce de detalles. Aquí también se afirma que no hay un estándar para el preprocesamiento de imágenes, como hemos podido observar en las diferencias de cada uno de estos papers. [15]
- **Evaluación del impacto del preprocesamiento de imágenes en la segmentación del iris:** En este trabajo se utilizan hasta 5 algoritmos diferentes para preprocesar imágenes para la segmentación del iris del ojo. Estos algoritmos están basados en el ajuste de histograma, filtros Gaussianos y, adicionalmente debido al problema concreto que se trata, la eliminación del reflejo especular del ojo, todo ello aplicando el método de segmentación. [43]

Una vez analizados estos trabajos, y como se afirma en el tercero, vemos que no hay un estándar de cómo preprocesar las imágenes a día de hoy, pero a su vez esto ha causado el avance y uso de distintas técnicas de las que ahora podemos disponer para preprocesar nuestras imágenes como más nos convenga. Sí que podemos observar elementos comunes como la eliminación de ruido o los filtros Gaussianos que nos convendría sentar como base para nuestro preprocesamiento de imágenes.

3 Indexado de Imágenes 2D

3.1. Qué es el indexado

La indexación de imágenes es el proceso de asignar etiquetas o metadatos a las imágenes para que puedan ser identificadas y recuperadas con mayor facilidad en un sistema de almacenamiento de imágenes. La indexación de imágenes es un paso importante en el procesamiento de datos de imágenes, ya que permite la búsqueda y recuperación eficiente de imágenes en grandes conjuntos de datos de imágenes. [10]

En el contexto del big data, la indexación de imágenes es particularmente importante, ya que los conjuntos de datos de imágenes pueden ser extremadamente grandes y complejos, lo que dificulta la búsqueda y recuperación de imágenes relevantes. La indexación de imágenes permite la creación de una base de datos de imágenes que puede ser fácilmente consultada para encontrar imágenes específicas basadas en sus características, como colores, formas, objetos, tamaño, entre otros. [4]

3.2. Por que hay que realizar la indexación

La indexación de imágenes es importante en el contexto actual por varias razones, estas podemos encontrarlas expuestas en los siguientes artículos [11, 25]:

- 1) Volumen de datos: Con el aumento del uso de dispositivos móviles, las redes sociales y el Internet de las cosas, se está generando una enorme cantidad de datos de imágenes. La indexación de imágenes es esencial para procesar y gestionar eficientemente grandes conjuntos de datos de imágenes.
- 2) Búsqueda eficiente: La indexación de imágenes permite la búsqueda y recuperación eficiente de imágenes relevantes en grandes conjuntos de datos de imágenes. Esto es especialmente importante en aplicaciones donde se necesita encontrar rápidamente una imagen específica entre millones de imágenes, como en la medicina, la seguridad y la vigilancia.
- 3) Análisis de datos: La indexación de imágenes también es útil para el análisis de datos de imágenes a gran escala, como en la industria de la publicidad, la detección de fraudes y la exploración de datos científicos.
- 4) Automatización: La indexación de imágenes también permite la clasificación automática de imágenes, lo que puede ahorrar tiempo y recursos en la gestión de grandes conjuntos de datos de imágenes.

3.3. Como se realiza

Teniendo en cuenta todos los aspectos anteriores, hay varias formas de realizar la indexación de imágenes 2D. La eficiencia de estas dependerá de las características de las imágenes y de la base de datos.

Las técnicas de indexación de imágenes pueden hacer uso de las características de las imágenes o de descriptores¹ de esas características para llevar a cabo la indexación. Por ejemplo, se puede hacer uso del tamaño, la textura o el color de la imagen para clasificarlas. También se puede hacer uso de redes neuronales para extraer características. Luego, una vez se han extraído las propiedades de las imágenes se pueden usar algoritmos de *clustering*, como *K-Means*, para agrupar imágenes similares en una misma categoría. En la figura 1 se puede observar una explicación a grandes rasgos de lo que hace este algoritmo, siendo los puntos nuestras imágenes y los círculos las categorías.

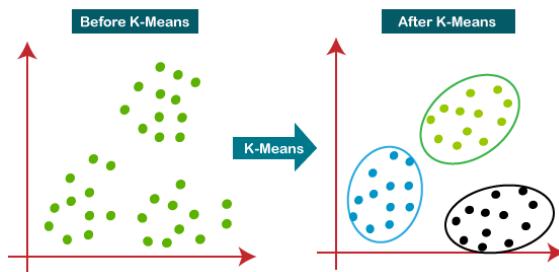


Fig. 1: K-Means Clustering

3.4. Estado del arte

Respecto al estado del arte del indexado de imágenes 2D, este se encuentra en una evolución constante, pues con la revolución de las IAs en los últimos tiempos se consigue un potencial enorme en este campo, entre muchos otros.

Hay varios papers que hablan sobre el tema. A continuación, se recogen varios de ellos en castellano:

- **Recuperación de imágenes sobre la base del contenido:** <https://tesis.ipn.mx/bitstream/handle/123456789/21079/XD%2007.13.pdf>
- **Recuperación de imágenes mediante rasgos descriptores globales y locales:** <https://repositoriodigital.ipn.mx/bitstream/123456789/7302/1/Tesis%2012350.pdf>
- **Recuperación por contenido en bases de datos de imágenes basada en wavelets: aplicación al diseño del textil:** https://www.researchgate.net/profile/Manuel-Agusti-Melchor/publication/267686624_Recuperacion_por_contenido_en_bases_de_datos_de_imagenes_basada_en_wavelets_aplicacion_al_diseño_del_textil/links/547d97090cf285ad5b08a72e/Recuperacion-por-contenido-en-bases-de-datos-de-imagenes-basada-en-wavelets.pdf
- **Técnicas de Recuperación por Contenido para Imagen y Vídeo en Arquitecturas Paralelas:** <https://oa.upm.es/182/1/10200416.pdf>

En inglés, varios papers que tratan el indexado de imágenes son:

¹Los descriptores de características son algoritmos que se utilizan para extraer información relevante de las imágenes y convertirla en una representación numérica. Algunos ejemplos de descriptores de características son SIFT, SURF, ORB, HOG, etc.

- **A New Descriptor for 2D Depth Image Indexing and 3D Model Retrieval:** https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4379599&casa_token=8EDNC9GIwe8AAAAA:3QR1m0uYEk30VFhFc9m6pUM3gy3Vj_x3XpZcMkKkU0IUDZtag=1
- **Color and Texture Features for Image Indexing and Retrieval:** https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4809223&casa_token=2NUXP4XMpt4AAAAA:W9BQ2uc3udxH93gRL-0PrSeZm2gDsecacByM4PFPIAC9ap1VUyCrIJsb140uAlPrbK6w&tag=1
- **Content-Based Image Indexing:** <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=56a59934a7c2a192c9c1068dd25c5af6cda642eb>
- **Structural High-resolution Satellite Image Indexing:** <https://hal.science/hal-00458685/document>
- **Image Indexing Using Color Correlograms:** https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=609412&casa_token=w4UfAryxuEIAAAA: NJ_3WMVl3uefh_3JE4vs-_6R0XWZhuAqE0i-U9bVmowBSZBk9K7PZVHyw6GYA9CD1Y-_IM733Q
- **Texture Based Image Indexing and Retrieval** <https://www.scitepress.org/PublishedPapers/2007/20658/20658.pdf>
- **Spectral Hashing With Semantically kConsistent Graph for Image Indexing** https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6202346&casa_token=70mKunMZJTYAAAAA:dzMCcXWU6V066jct0xDYkaEsj4qH4ysPorh7Tp_p0Vs5XH2R4IPAtCX3_3aHRkvuCYpUmoDsWA

Reconstrucción 3D

En este capítulo se plantea tratar el problema de la obtención de un modelo 3D a partir de una serie de imágenes 2D, para ello se llevará a cabo un proceso que consta primeramente de la extracción de las características de la imagen. Seguido de un cálculo de correspondencias y una estimación de la posición de la cámara, así como su reconstrucción en estéreo y visualización.

4 Introducción

4.1. Motivación

La capacidad de modelar entornos en tres dimensiones a partir de fotografías facilita la tarea de conservación del patrimonio cultural [29] así como la visualización de la evolución de los monumentos a lo largo del tiempo.

4.2. Contexto

La **fotogrametría** es el campo que reúne a todas la técnicas que son capaces de transformar fotografías en modelos tridimensionales. En este proyecto se desarrollará un método que a partir de una colección de fotografías realice una reconstrucción 3D utilizando técnicas de fotogrametría. Una reconstrucción 3D consiste en la obtención de un modelo tridimensional compuesto por la unión de imágenes captadas desde múltiples vistas.

El proceso de una reconstrucción 3D a partir de un conjunto de fotografías puede dividirse en las siguientes fases: Adquisición, Extracción de características, Cálculo de Correspondencias, Estimación de los Extrínsecos/Intrínsecos de las cámaras, Reconstrucción Estéreo y Aplicación.

La fase de **adquisición** consiste en la obtención de fotografías desde múltiples localizaciones de un mismo objeto u entorno. Tras su obtención se realiza un preprocesado antes de almacenar las imágenes en una base de datos o de manera estructurada para su posterior uso en las siguientes fases del proceso.

Tras la fase de adquisición se extraen las **características** de los puntos de las imágenes, estas características pueden ser usadas para obtener un conjunto localizaciones correspondientes en diferentes imágenes [39]. Con la enorme cantidad de datos redundantes presentes en la imágenes, la extracción de características es un proceso fundamental [32]. Los puntos clave (o *Keypoints* en inglés) son características que pueden ser detectadas a pesar de los cambios en la imagen de: resolución, escalado, iluminación, ruido, orientación y perspectiva [28]. Para detectar los *Keypoints* se utilizan métodos detectores de características (*Feature Detectors* en inglés) que además de detectar generan un descriptor de punto (*Feature Descriptor* en inglés). Dichos descriptores resumen la estructura local alrededor del punto al que describen. También existen descriptores globales que permiten comparar imágenes por completo entre si en lugar de comparar puntos individuales.

Una vez obtenidas las características de las imágenes, deberemos tener en cuenta las correspondencias, también conocido como **Matching**. Encontrar correspondencias entre pares de imágenes es una tarea fundamental en la visión artificial,

con aplicaciones que van desde la calibración de la cámara hasta el flujo óptico, Estructura de Movimiento (SfM), localización visual, seguimiento de puntos y la estimación de la pose humana. Actualmente, existen dos métodos que abordan este problema. Uno consiste en extraer conjuntos de puntos clave escasos de ambas imágenes y unirlos en orden para minimizar una métrica de alineación, mientras que el otro consta de interpretar la correspondencia como un proceso denso, donde cada píxel de la primera imagen se asigna a un píxel de la segunda. Es decir, su objetivo principal es el de obtener unos datos característicos los cuales coincidan o sean similares en ambas imágenes. De esta forma, podremos conseguir un *matching* entre dichas imágenes para posteriormente trabajar con los puntos claves [20].

La siguiente fase consistiría en **estimar** los parámetros de la cámara. Recuperar la orientación de una imagen es uno de los aspectos fundamentales en los problemas de visión por computador. La estimación precisa de la posición de la cámara es una de las claves a la hora de aplicarla en la reconstrucción de edificios 3D [35]. Debido a su importancia práctica, se ha dedicado mucho esfuerzo de investigación al tema a lo largo de los años. Un enfoque popular del problema se basa en detectar y hacer coincidir puntos de características locales y usar las correspondencias obtenidas para determinar las poses relativas. Para ello debemos también tener en cuenta que el rendimiento depende en gran medida de la precisión de las coincidencias de las características[26].

Por ultimo tenemos la **visualización** de la imagen en 3D. En ella se emplea un software de gráficos por ordenador para crear una imagen tridimensional y simular una percepción de profundidad por parte del espectador.

4.3. State of Art

4.3.1. Extracción de características

El primer enfoque de los algoritmos de extracción de características fueron los métodos **detectores de esquinas** (Förstner [14], Harris [16], Shi-Tomasi [36]). Entre estos métodos se encuentra SUSAN [37] que se basa en obtener la cantidad de vecinos de un píxel que tienen intensidad similar y tras aplicar un umbral los mínimos locales se clasifican como esquinas. De manera similar en FAST [33] un punto es clasificado como una esquina si existe una gran cantidad de píxeles significativamente más brillantes en un radio a su alrededor. Otros enfoques también populares son la **transformada de Hough (HT)** y la **diferencia de gaussianas (DoG)**.

Los métodos descriptores de características obtienen vectores (denominados descriptores) que describen la estructura local alrededor de un punto clave. El algoritmo SIFT [24] utiliza DoG para obtener los puntos clave y sus resultados son bastante robustos. Debido a la alta carga computacional de SIFT se propone SURF [5], que aunque reduce la carga computacional también los hace su robustez. SURF utiliza la matriz Hessiana para obtener los puntos clave.

El descriptor de SIFT es altamente discriminante al ser un vector de 128 elementos pero esto causa que su calculo y *matching* sean relativamente lentos de computar. Esto puede una problemática para aplicaciones de tiempo real que requieran una gran cantidad de descriptores como por ejemplo la reconstrucción 3D a gran escala. Para solucionar esto surge BRIEF [7]. También basándose en BRIEF surge ORB [34] que obtiene resultados dos ordenes de magnitud más rápidamente sin comprometer su robustez.

Aunque BRIEF tenga una ventaja en velocidad su tolerancia a distorsiones y ciertas transformaciones es mínima, lo que requiere del uso de métodos probabilísticos para obtener el consenso a la hora de realizar el *matching* entre descriptores. Para mejorar este aspecto se plantea BRISK [23] que utiliza FAST para la detección de puntos clave.

En [3] se propone la agregación de los algoritmos de SIFT, SURF y ORB mediante el método de *Random Forest* lo que produce mejora los resultados en tareas como el reconocimiento de objetos. Adicionalmente mencionar que existen otros métodos que utilizan técnicas de *Machine Learning* como los basados en *deep learning* para la extracción de características [21].

4.3.2. Matching: Cálculo de Correspondencias

El enfoque ingenuo sería utilizar una métrica que tome como entrada los píxeles sin procesar de las imágenes y genere su similitud. Es habitual pensar en el uso del Mean Squared Error (MSE - error cuadrático medio) para calcular el valor medio del cuadrado de las diferencias entre los píxeles de las dos imágenes. Sin embargo, a pesar de su velocidad y simplicidad, este método presenta muchos problemas, ya que una gran distancia euclidiana entre las intensidades de píxeles no significa necesariamente que el contenido de la imagen sea diferente. Como podemos ver a continuación 2, si solo cambiamos el contraste de la imagen, el valor final de MSE aumentará mucho, aunque no cambiamos el contenido:

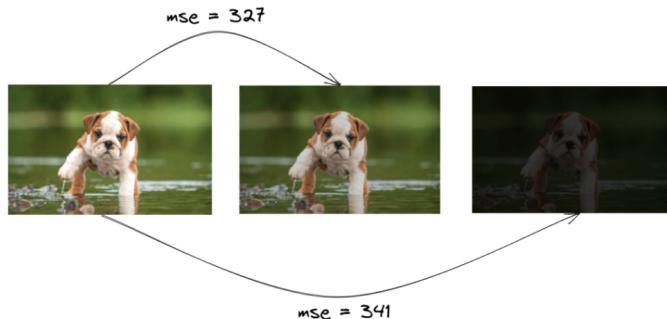


Fig. 2: Variación del MSE [2]

Los métodos basados en píxeles como MSE no son efectivos cuando las imágenes de entrada se toman bajo diferentes ángulos o condiciones de iluminación, además de la posible desventaja del tiempo de ejecución. Para hacer frente a estos casos, los métodos de *Matching* son los más adecuados. En estos tipos de algoritmos encontramos algunos como el algoritmo quadtrees o partición de espacio binario (BSP).

Como se ha mencionado anteriormente con el método MSE, la complejidad computacional cuadrática plantea un obstáculo importante para aplicarlos a tareas de visión que requieren predicciones densas, como detección de objetos, coincidencia de características, estéreo, etc. El algoritmo QuadTree, reduce la complejidad computacional cuadrática a lineal, construyendo pirámides de tokens (Figura 3). [40]

Este algoritmo consta de dos procesos principales, el proceso de partición binaria repetitiva y el proceso de coincidencia. En primer lugar, en el proceso de partición binaria, el algoritmo realiza recursivamente la partición binaria que divide

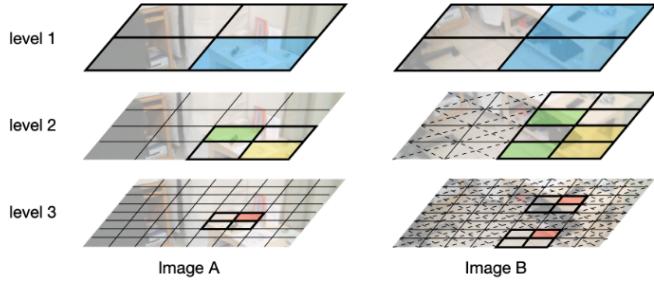


Fig. 3: Niveles de la pirámide del algoritmo quadtree

las regiones en dos particiones que cubren por completo esas regiones. Este proceso de subdivisión da lugar a una representación de objetos dentro del espacio en forma de una estructura de datos de árbol conocida como árbol BSP (Figura 4). En segundo lugar, se lleva a cabo el proceso de emparejamiento. Este algoritmo promete una sobrecarga computacional baja, ya que calcula fácilmente la intersección entre regiones en los límites de partición y no requiere comparaciones innecesarias. [1]

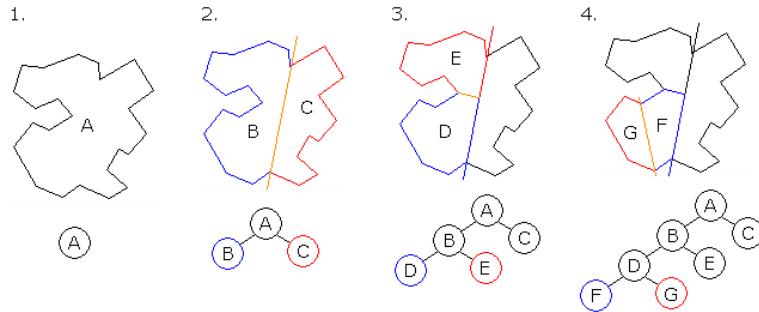


Fig. 4: Estructura de un árbol BSP (Binary Space Partitioning)

4.3.3. Camera Pose Estimation

El enfoque para la estimación de la pose de la cámara es a partir de un conjunto de coincidencias 2D-3D entre píxeles en una imagen de consulta y puntos 3D en una escena modelo. El procedimiento clásico seguido en la estimación de la pose basada en la estructura consiste en primero obtener un conjunto de coincidencias 2D-3D. Estas coincidencias generalmente se obtienen al hacer coincidir puntos clave en una imagen de consulta (o secuencia de imágenes) contra descriptores medios asociados a cada punto 3D en el modelo. Dadas estas coincidencias, la pose de la cámara extrae una conclusión mediante el empleo de un solucionador mínimo dentro de un esquema de hipótesis y verificación [8].

Con el anterior procedimiento surge un problema y es que está limitado a la calidad de los puntos 3D que la cámara observa. Por lo tanto otra forma de enfocarlo sería obteniendo coincidencias entre imágenes 2D. En estos casos, la consulta generalmente se compara con las imágenes más similares. Alternativamente, también se pueden obtener coincidencias 2D-2D siguiendo el mismo procedimiento descrito para la estimación de la pose basada en la estructura. Una vez que se obtiene un

conjunto de coincidencias 2D-3D, se puede reemplazar la parte 3D. A continuación, se utiliza un solucionador mínimo dentro de un bucle RANSAC con el fin de estimar de forma robusta una pose de cámara. Dependiendo de si cámara está calibrada o no, se necesitarán más puntos coincidentes [8].

Como podemos observar en la Figura 5 se encuentran los dos métodos comentados anteriormente.

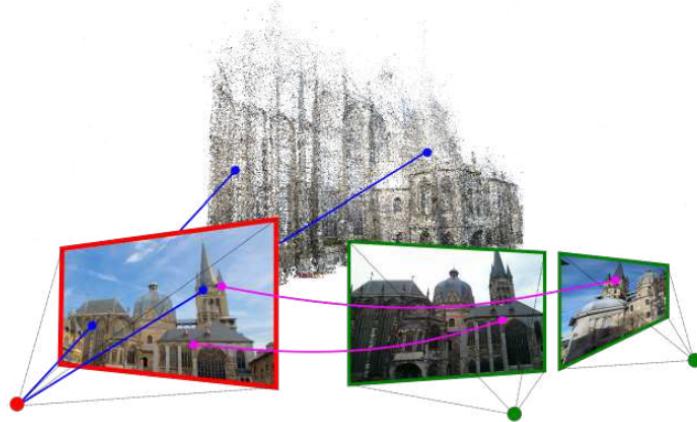


Fig. 5: Visualización de coincidencias 2D-2D (rosa) y 2D-3D (azul)

En los últimos años, las redes neuronales convolucionales profundas (dCNNs) han demostrado un alto éxito en el aprendizaje de diferentes tareas de visión artificial que nos podrían ayudar a desarrollar esta fase. Aprovechando la idea del aprendizaje por transferencia, las dCNN preentrenadas se usaron más para aprender otras tareas visuales que tienen una cantidad limitada de datos de entrenamiento pero comparten las mismas características de bajo nivel. Motivados por estos avances se propuso PoseNet [22] que fue la primera arquitectura basada en el aprendizaje que introdujo la idea de hacer retroceder la pose absoluta con una arquitectura profunda. Este enfoque ofrecía varias ventajas atractivas en comparación con métodos clásicos basados en estructuras, tiempo de inferencia cortos y huella de memoria baja [35].

4.3.4. Reconstrucción Estéreo

En este proceso, la solución clásica, reconstruye la estructura de un objeto en un conjunto 3d a partir de dos o más imágenes 2d del mismo, de las que se conoce el punto de vista de la cámara, de forma que permita triangular los puntos 3d con la intersección de rayos proyectados desde las cámaras. Tras tratar la correspondencia por matching, se intenta obtener un punto 3d por cada par de puntos correspondientes, buscando el punto medio entre las proyecciones. El uso de más de dos imágenes de referencia se conoce como Multi-view stereo(MVS).

Los algoritmos estéreo sufren por la falta de textura superficial local debido a la restricción de la suavidad de profundidad o desajustes locales en las estimaciones de disparidad. Por lo tanto, la mayoría de los métodos estéreo sólo proporcionan un mapa de profundidad aproximado que se puede asociar con una imagen suavizada del mapa de profundidad. Por otro lado, los algoritmos Shape From Shading (SFS), generalmente producen mejores estimaciones de áreas superficiales locales, pero

algunos de ellos tienen problemas con el albedo variable y superficies esféricas. Por lo tanto, los métodos SFS producen una información de profundidad detallada mejorada.

Debido a esto, ha habido aplicaciones en las que se han combinado los métodos como en [9]. En este caso, se utiliza la reconstrucción estéreo por binocular para obtener la superficie general del objetivo. Por otro lado, el método de Shadow Recovery Shape puede recuperar con precisión la información detallada de la textura de la superficie objetivo. Tras esto se aplica el algoritmo conjunto de las técnicas.

Con el auge del Machine Learning(ML), también ha llevado a el entrenamiento de un modelo que mejoren los resultados, como se puede ver en [30], donde se utiliza una solución estéreo convencional, partiendo de dos imágenes, para obtener una un mapa de profundidad aproximado y después aplican el modelo entrenado como corrector sobre el resultado anterior, mejorándolo. También incluye una solución para su aplicación con una imagen(ResDepth-mono), con la desventajas que ello conlleva.

4.3.5. Visualización

La visualización de imágenes en 3D por ordenador es un proceso en el que se utiliza un software de gráficos por ordenador para crear una imagen tridimensional y representarla en una pantalla o monitor en una forma que simula una percepción de profundidad para el observador.

Para crear una imagen en 3D por ordenador, se utilizan técnicas de modelado, iluminación, texturización y animación para crear una escena tridimensional virtual. Luego, se utiliza un proceso de renderizado para generar una imagen bidimensional que simula una vista desde un punto de vista específico. Esta imagen se muestra en la pantalla o monitor del ordenador.

5 Marco Teórico

5.1. Pipeline

En esta sección se desarrollarán de manera teórica las fases del proceso de modelización 3D. Estas fases están basadas en los modelos descritos en las figuras 6a y 6b, combinando dichos modelos planteamos las siguientes fases para la reconstrucción 3D:

- 1) Cálculo de las correspondencias entre todas las imágenes de la colección.
- 2) Estimación simultánea de los parámetros de orientación intrínsecos y extrínsecos de la cámara mediante *Bundle Adjustment* [42]. Este paso produce una nube de puntos poco densa.
- 3) Densificación de la nube de puntos del paso anterior mediante *Multiview Stereoopsis (MVS)*.
- 4) Creación de un modelo de malla 3D.
- 5) Visualización del modelo de malla.

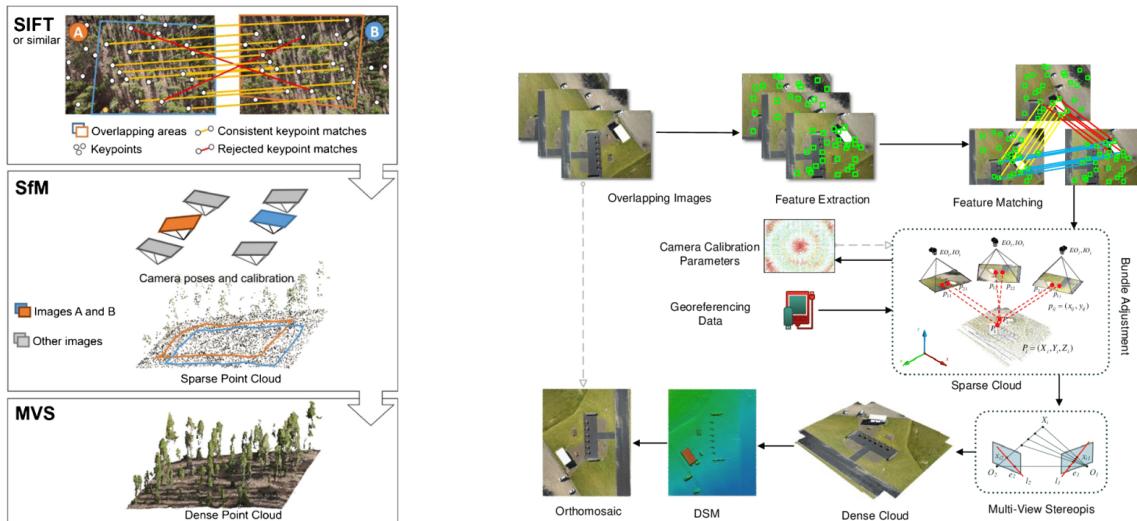


Fig. 6: Fases del proceso de obtención de nubes de puntos a partir de fotografías con superposición y un ejemplo de aplicación del proceso.

5.2. Descriptores y correspondencias

En esta fase se calcularán las correspondencias siguiendo el esquema propuesto en [38]. Como primer paso del proceso se extraen los descriptores de las imágenes mediante el detector de puntos clave SIFT [24], debido a su reducida varianza ante

transformaciones en las imágenes. A continuación por cada par de imágenes de la colección se asocian los descriptores mediante el algoritmo de *Approximate Nearest Neighbors* de la siguiente manera:

- 1) Se construye un *kd-tree* [6] con los descriptores de la primera imagen.
- 2) Se obtienen los dos vecinos más cercanos de cada descriptor de la segunda imagen en el árbol creado en el paso anterior.
- 3) La correspondencia entre un descriptor de la segunda imagen y su vecino más cercano en la primera, se acepta si la relación entre la distancia del descriptor con su primer y segundo vecino cumple que: $\frac{d_1}{d_2} > 0,6$.
- 4) Si existe un descriptor con más de una correspondencia entre dos imágenes, se eliminan todas con las que este relacionado.

Una vez obtenidas las correspondencias iniciales, se estima la matriz fundamental mediante el algoritmo de ocho puntos [17] utilizando un esquema iterativo Random Sample Consensus (RANSAC) [12] con un umbral de valores atípicos (en inglés *outlier threshold*) del 0.6 % del tamaño máximo de la imagen. Tras esto se refina la matriz fundamental mediante el algoritmo de optimización de **Levenberg-Marquardt**. Utilizando el mismo umbral que con RANSAC, se eliminan las correspondencias que sean valores atípicos respecto a la matriz fundamental, si alguna imagen tiene menos de 20 correspondencias no se tendrán en consideración para el siguiente paso.

Por último se ordenan las correspondencias en **tracks**, que son un conjunto de puntos clave correspondientes transversalmente en múltiples imágenes. Si un *track* tiene más de un punto clave en una imagen se considera inconsistente. Manteniendo todos los *tracks* consistentes y con al menos dos puntos clave ya se tiene la información suficiente para pasar a la siguiente fase.

5.3. Bundle Adjustment

5.3.1. ¿Qué es?

Bundle Adjustment es el problema de refinrar una reconstrucción visual para producir estimaciones óptimas conjuntas de la estructura 3D y los parámetros de visualización (posición y calibración de la cámara). Con óptimas nos referimos a que las estimaciones de los parámetros se encuentran minimizando alguna función de costo que cuantifica el error de ajuste del modelo y, en conjunto, que la solución es simultáneamente óptima con respecto a la estructura y las variaciones de la cámara. Es un problema de estimación de parámetros geométricos dispersos, los parámetros son las coordenadas de características 3D combinadas, las poses de la cámara y las calibraciones. El nombre se refiere a los haces de rayos de luz que salen de cada función 3D y convergen en cada centro de la cámara, que se ajustan de manera óptima con respecto a las posiciones de la función y la cámara [41].

5.3.2. ¿Cómo se obtiene?

Para la reconstrucción incremental del modelo 3D se siguen ciertos pasos, los cuales son, inicialización, registro de imagen, triangulación y por último bundle ad-

justment que es en el paso que nos vamos a centrar. Como podemos ver en la Figura 7 así es como funcionaría.

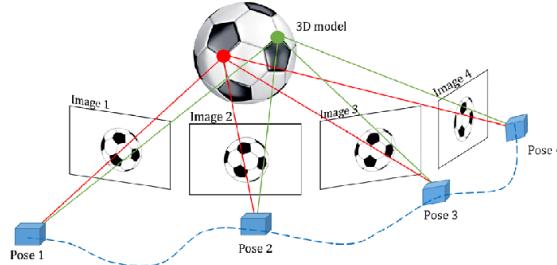


Fig. 7: Ejemplo ilustrado del Bundle Adjustment

Para ello vamos a necesitar los siguientes valores con el fin de poder refinar la reconstrucción. Una matriz de 3×3 con los puntos 3D sin refinar, en estas se encuentran las localizaciones x y z. Los puntos que coinciden en múltiples imágenes, especificado como un array de N elementos con objetos pointTrack, cada uno de estos elementos contiene dos o más puntos coincidentes a través de múltiples imágenes. La información de la pose de la cámara, especificado como una tabla de dos columnas, una con la pose absoluta y otra con el id de la vista. La vista de las id está relacionada con las id de los objetos comentados anteriormente. Por último, los intrínsecos de la cámara especificados como objetos o un array de N elementos con estos objetos. N sería el número de poses de la cámara o el números de ids.

MIRAR METODO estimateCameraParameters() PARA OBTENER LOS DATOS DEL ARCHIVO cameraParams.mat

5.3.3. ¿Qué obtenemos?

Después de utilizar el método de Matlab bundleAdjustment, entre los tipos de resultados que puede devolver este método, nos interesan especialmente tres:

- **xyzRefinedPoints:** Uno de los primeros resultados son los puntos estimados en un espacio de 3 dimensiones pero refinados por el método y devueltos en forma de matriz de $M \times 3$ para las localizaciones.
- **refinedPoses:** Igual que se han refinado los puntos, también se refinan las poses de la cámara en una tabla de 2 columnas, siendo estas para el ViewId y AbsolutePose, como las cameraPoses pasadas por parámetro.
- **reprojectionErrors:** Para cada uno de los puntos que se ha proyectado, se calculará el error medio de proyección con la diferencia de distancia entre los puntos detectados y los proyectados y se almacenará en este vector.

Finalmente, con este resultado, podremos construir una nube de puntos 5.5 precisa para las imágenes que hemos recibido8.

5.4. Multiview Stereopsis (MVS)

Multiview Stereopsis es un proceso basado en tomar varias imágenes desde distintas perspectivas de un objeto con el fin de calcular la posición 3D de sus puntos en el espacio. Una vez, se han obtenido estos puntos, se les aplica una textura con

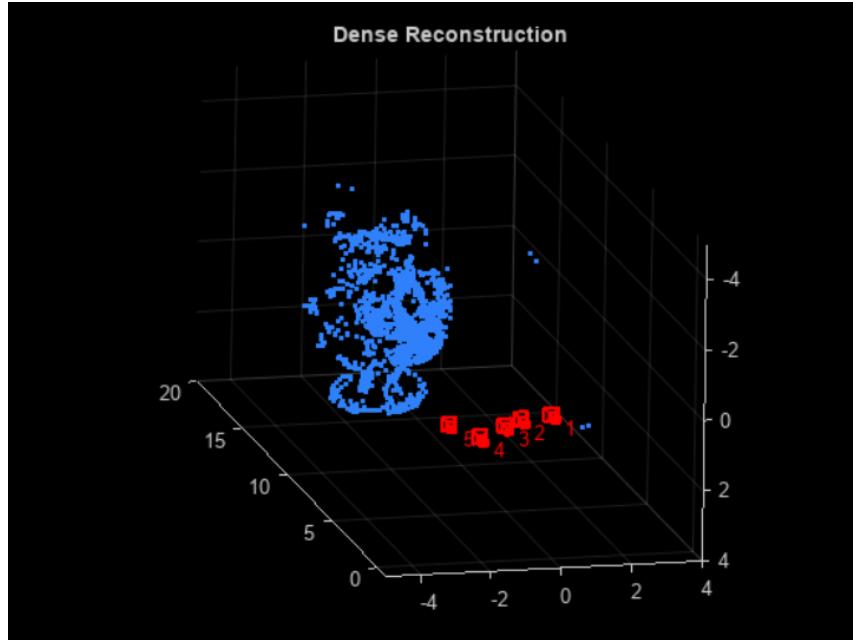


Fig. 8: Nube de puntos

el fin de hacerlos más afines a la realidad. La MVS generalmente implica varios pasos, que incluyen adquisición de imágenes, extracción de características, emparejamiento, triangulación y reconstrucción 3D. El proceso puede ser exigente en términos de cómputo, especialmente cuando se trabaja con grandes cantidades de imágenes. Cabe resaltar que la MVS no solo puede usarse para modelado 3D (como será nuestro caso) sino que también se puede utilizar para la robótica, para la realidad virtual entre otras muchas cosas.

5.5. Nube de puntos

5.5.1. ¿Qué es?

Una nube de puntos 3D es el primer producto resultante del escaneo láser o la fotogrametría digital y el paso previo a la creación de modelos 3D. Se compone por millones de puntos posicionados de forma tridimensional en el espacio, formando con exactitud una entidad física y representando su superficie externa. La nube de puntos 3D contiene una amplia información métrica sobre las superficies escaneadas, así como la relativa a su color y reflectividad del material. Gracias a una gran cantidad de puntos situados en el espacio y con una representación del color existente en las superficies escaneadas, se reproduce el espacio analizado a través de una geometría de gran precisión. Por lo tanto, el documento obtenido se puede entender como una foto 3D del espacio analizado, con una gran exactitud en sus dimensiones. Con esta recogida de datos obtenemos una gran fiabilidad, evitando los errores que son fáciles de cometer con métodos tradicionales. Además, vamos a poder recurrir a la información recogida en el archivo de escaneado siempre que sea necesario, teniendo una imagen 3D fiable del edificio o infraestructura tal y como se encontraba en la fecha en la que se hizo la recogida de datos.

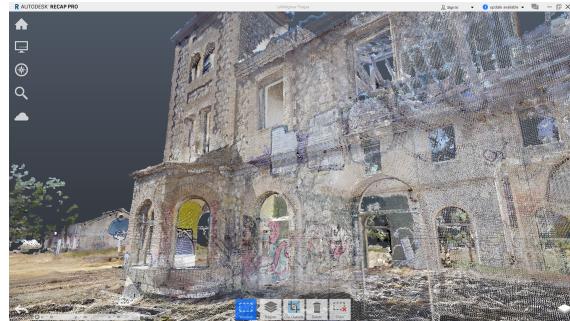


Fig. 9

5.5.2. ¿Cómo se obtiene?

Existen diferentes maneras de obtener una nube de puntos. Aunque el resultado final siempre va a resultar una malla de puntos 3D de color, las diferencias entre unos escaneados y otros van a estar en el equipo y tecnología usados para conseguir este resultado.

- **Escaner láser 3D:** es un dispositivo de adquisición de datos masivos, que nos reporta una nube de puntos generada tridimensionalmente, a partir de la medición de ángulos y distancias, mediante un haz de luz láser.
- **Fotogrametría digital:** es aquella que utiliza múltiples imágenes de un objeto desde diferentes ángulos, para generar una nube de puntos 3D del objeto de gran calidad métrica.

5.5.3. ¿Qué se puede generar?

- **Recursos multimedia y realidad virtual:** El nivel de realismo que graban los últimos dispositivos de escáner láser con imágenes HDR, nos permiten generar recursos multimedia de calidad para comunicar diferentes tipos de información, desde imágenes realistas del estado actual de un entorno u objeto a vídeos de recorrido virtual por las nubes de puntos.
- **Ortoimagen:** Se trata de una representación bidimensional de imagen (jpg, png, bmp, etc.) de una vista ortogonal del entorno escaneado, donde cada elemento representado se encuentra en verdadera magnitud, teniendo la misma validez que una medida en un plano clásico. Por tanto, podemos tomar medidas reales sin estar en el lugar *in situ*. Además, podemos generar ortoimágenes por cada plano de interés, ya sean, alzados o secciones, éstas últimas muy utilizadas para análisis geométricos o de deformaciones.
- **Mallas 3D:** Esta malla se considera como un proceso secundario tras la nube de puntos, nos permite pasar de un modelo 3D formado únicamente por puntos, a un modelo 3D de malla formado por pequeños planos poligonales (triángulos). Esta malla se trata de un objeto cerrado que se adapta a la forma real del objeto medida en la nube de puntos. Se profundizará acerca de este punto en la siguiente sección.

5.6. Visualización del modelo de malla 3D

Para poder ver los modelos obtenidos, el último paso, tenemos dos opciones: integrar aplicaciones de visualizado 3D ya existentes en nuestra aplicación, por ejemplo Paraview; o utilizar librerías de lenguajes como pyvista en python, que permitan dibujar la malla, para implementarlo desde cero directamente.

Plan de Financiación

6 ...

Plan de Mantenimiento

7 ...

Siglas

RANSAC Random Sample Concensus. 19

SIFT Scale-invariant feature transform. 7, 13, 14, 18

Referencias

- [1] November 2011. [Online; accessed 6. Mar. 2023]. 15
- [2] Algorithms for Image Comparison | Baeldung on Computer Science, November 2022. [Online; accessed 6. Mar. 2023]. 14
- [3] Monika Bansal, Munish Kumar, and Manish Kumar. 2d object recognition: a comparative analysis of sift, surf and orb feature descriptors. *Multimedia Tools and Applications*, 80:18839–18857, 5 2021. 14
- [4] José Orlando Maldonado Bautista. Estudio de métodos de indexación y recuperación en bases de datos de imágenes. *Universidad del País Vasco*, 2008. 9
- [5] H Bay, Tinne Tuytelaars, Luc Van Gool, A Leonardis, H Bischof, and A Pinz. Surf: Speeded up robust features. *Lecture Notes in Computer Science*, 3951:404–417, 1 2006. 13
- [6] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18:509–517, 9 1975. 19
- [7] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6314 LNCS:778–792, 2010. 13
- [8] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid camera pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 15, 16
- [9] Ziyi Cao, Yaxiang Wang, Wenfeng Zheng, Lirong Yin, Yushan Tang, Wang Miao, Shan Liu, and Bo Yang. The algorithm of stereo vision and shape from shading based on endoscope imaging. *Biomedical Signal Processing and Control*, 76:103658, 7 2022. 17
- [10] Tzi-cker Chiueh. Content-based image indexing. In *VLDB*, volume 94, pages 582–593. Citeseer, 1994. 9
- [11] Heting Chu. Research in image indexing and retrieval as reflected in the literature. *Journal of the American Society for Information Science and Technology*, 52(12):1011–1018, 2001. 9
- [12] Martin A. Fischler and Robert C. Bolles. Random sample consensus. *Communications of the ACM*, 24:381–395, 6 1981. 19
- [13] Alejandro N. Florencia Ysiquio. Modelado de sistemas de control de un robot manipulador basado en procesamiento digital de imágenes. 2004. Accessed: 2023-03-08. 7
- [14] Wolfgang Förstner and Eberhard Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. volume 6, pages 281–305, 1987. 13
- [15] Juan P. Graffigna. Procesamiento digital de imágenes. 2003. Accessed: 2023-03-08. 8

- [16] Chris Harris, Mike Stephens, et al. A combined corner and edge detector. volume 15, pages 10–5244, 1988. 13
- [17] Richard I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:580–593, 1997. 19
- [18] Jakob Iglhaut, Carlos Cabo, Stefano Puliti, Livia Piermattei, James O'Connor, and Jacqueline Rosette. Structure from motion photogrammetry in forestry: a review. *Current Forestry Reports*, 5:155–168, 2019. 18
- [19] Farid Javadnejad. Small unmanned aircraft systems (uas) for engineering inspections and geospatial mapping, 4 2018. 18
- [20] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. COTR: Correspondence Transformer for Matching Across Images, 2021. [Online; accessed 2. Mar. 2023]. 13
- [21] Manjunath Joggin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva. Feature extraction using convolution neural networks (cnn) and deep learning. *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018 - Proceedings*, pages 2319–2323, 5 2018. 14
- [22] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 16
- [23] Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. *Proceedings of the IEEE International Conference on Computer Vision*, pages 2548–2555, 2011. 14
- [24] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 11 2004. 13, 18
- [25] Antoine Manzanera and Unité d'Électronique et d'Informatique. Indexación de imágenes y videos. Disponible online en http://www.enstaaristech.fr/~manzaner/Download/Tutorials/Curso_Indexacion_Bogota08.pdf Fecha último acceso, 1(08):2013, 2008. 9
- [26] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. *Relative Camera Pose Estimation Using Convolutional Neural Networks*. 2017. 13
- [27] Leonardo Mereles. Preprocesamiento de imágenes digitales a través de su transformada de fourier. 2012. Accessed: 2023-03-08. 8
- [28] Inc MicroImages. Raster image processing: Keypoint matching for image registration. 2015. Accessed: 2023-03-01. 12
- [29] Nikolaos, Protopapadakis Eftychios, Voulodimos Athanasios, Ioannides Mari-nos Doulamis Anastasios, and Doulamis. 4d modelling in cultural heritage. *Advances in Digital Cultural Heritage*, pages 174–196, 2018. 12
- [30] Corinne Stucker Konrad Schindler Photogrammetry, Remote Sensing, and Eth Zurich. Resdepth: Learned residual stereo reconstruction. pages 184–185, 2020. 17

- [31] Zambrano Martínez Jorge L. Plaza Cordero, Andrea M. Estudio y selección de las técnicas sift, surf y asift de reconocimiento de imágenes para el diseño de un prototipo en dispositivos móviles. 2012. Accessed: 2023-03-08. 7
- [32] Ben Robbins and Robyn Owens. 2d feature detection via local energy. *Image and Vision Computing*, 15:353–368, 5 1997. 12
- [33] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3951 LNCS:430–443, 2006. 13
- [34] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. *Proceedings of the IEEE International Conference on Computer Vision*, pages 2564–2571, 2011. 13
- [35] Yoli Shavit and Ron Ferens. Introduction to Camera Pose Estimation with Deep Learning. *arXiv e-prints*, page arXiv:1907.05272, July 2019. 13, 16
- [36] Jianbo Shi and Carlo Tomasi. Good features to track. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994. 13
- [37] Stephen M Smith and J Michael Brady. Susan—a new approach to low level image processing. *International journal of computer vision*, 23:45–78, 1997. 13
- [38] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International journal of computer vision*, 80:189–210, 2008. 18
- [39] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 12
- [40] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. QuadTree Attention for Vision Transformers. *arXiv*, January 2022. 14
- [41] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment — a modern synthesis. In Bill Triggs, Andrew Zisserman, and Richard Szeliski, editors, *Vision Algorithms: Theory and Practice*, pages 298–372, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg. 19
- [42] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. pages 298–372, 2000. 18
- [43] Poveda-Sendales Daniel A. Valencia-Vargas Daniel F. Valencia-Murillo, José F. Evaluación del impacto del preprocesamiento de imágenes en la segmentación del iris. 2014. Accessed: 2023-03-08. 8