
RECONSTRUCCIÓN DE OBJETOS 3D MEDIANTE MÚLTIPLES IMÁGENES

M. en C. Álvaro Anzueto Ríos

aanzueto@ipn.mx

Eduardo Orozco Pérez

edros.heart@gmail.com

Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas-IPN.

Resumen

En este artículo se describen los resultados obtenidos de los experimentos realizados con librerías disponibles de la reconstrucción 3D mediante múltiples imágenes. La paquetería usada está programada en un ambiente Python que nos permite tener seguimiento del proceso usado además de modificarlo y adaptarlo a las necesidades del usuario. El programa OSM-Bundler usa el algoritmo SIFT y el método MVS de reconstrucción para generar modelos con la fidelidad necesaria para ser ocupa en otras tareas, usando cámaras digitales con capacidad de capturar imágenes de alta definición. En el artículo se mencionan los aspectos técnicos de las cámaras usadas.

1.- Introducción

Hoy en día se ha incrementado la necesidad de poder crear modelos digitales de objetos de la vida real, sin embargo el costo de la creación del mismo ha hecho que el proceso sea difícil de adquirir individualmente, por algún proyecto personal o escolar, o de modo empresarial o institucional, por la reconstrucción de algún objeto para su modificación o rediseño.

El método de reconstrucción 3D usando múltiples imágenes es un método que cualquier persona con una cámara digital y una computadora puede realizar. El software fue desarrollado de manera libre, en lenguaje Python, y con cálculos usados para la visión artificial y el reconocimiento de patrones tales como el uso del algoritmo Scale-Invariant Feature transform (SIFT) [1], el método de Struction from Motion (SfM) y el método de reconstrucción Patch-based Multi-view Stereo Software (PMVS) que son parte del sistema de cálculo de movimiento Bundler [2]; cuyos principios se describirán más adelante dentro de este artículo, dando como resultado un modelo de resolución variable, ya que depende de la cámara y de la cantidad de imágenes usadas, fácil de manipular y con la capacidad de exportarlo a cualquier otro software de edición y modelado 3D [3, 4, 5].

2.- Desarrollo

Para reconstruir un objeto con el método de múltiples imágenes es necesario realizar ciertos procedimientos. Primero, para asegurar una reconstrucción exitosa algunas variables no fueron consideradas dentro de los cálculos.

En el siguiente diagrama se mencionan los elementos del proceso de una reconstrucción exitosa.

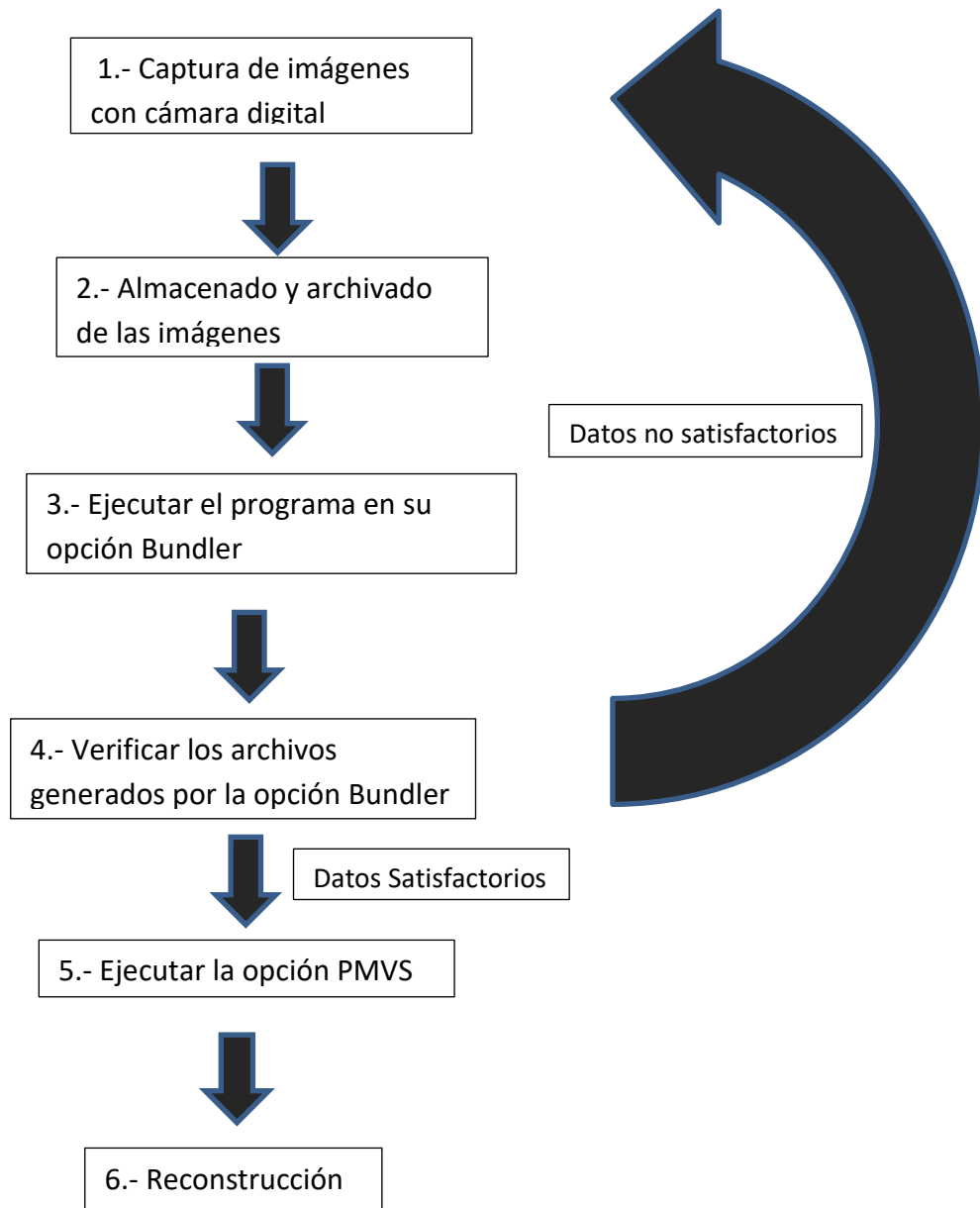


Diagrama 1. Proceso completo de reconstrucción en 3D

Cada uno de los elementos del Diagrama uno serán explicados con más detalle a continuación:

1. Captura de Imágenes: La captura de las imágenes del objeto debe ser realizado con una cámara digital puesto que el sensor de estas cámaras tiene la capacidad necesaria para que los cálculos se lleven a cabo con la menor cantidad de errores posible. Las capturas deben tener como centro de la toma el objeto desde varios ángulos. Se recomienda tomar por lo menos 12 imágenes del objeto, variando el ángulo de perspectiva. La superficie donde se coloque el objeto debe ser opaca para mejores resultados.
2. Almacenado y Archivado de Imágenes: Las imágenes tomadas varían en resolución dependiendo de la cámara lo cual afecta la calidad de la reconstrucción. El formato de las imágenes debe ser en .jpg y archivar todas las imágenes con nombres secuenciales en una sola carpeta con nombre alusivo al objeto.
3. Ejecutar el Programa en su Opción Bundler: El programa osm-bundler se ejecuta de forma manual en la ventana de comando, desde la carpeta donde este almacenado mediante el comando Python para así escribir el nombre de la opción RunBundler.py.
4. Verificar los Archivos Generados: una vez terminada la ejecución de la opción Bundler, el programa genera una serie de archivos relevantes para la reconstrucción, dentro de los mismos hay un archivo de texto, .txt, el cual contiene el nombre de las imágenes seguido de una serie de números. Para saber que los archivos son satisfactorios la serie de números debe empezar con 0 y todas las secuencia deben ser la misma.
5. Ejecutar la Opción PMVS: Para ejecutar la opción PMVS se realiza el mismo procedimiento que para bundler solo que se escribirá la ubicación de la carpeta que contiene los archivos generados por Bundler.
6. Reconstrucción: terminado el proceso de reconstrucción, la opción PMVS generará una carpeta ubicada dentro de la carpeta de los archivos del creados por el Bundler llamada PMVS, dentro hay más carpetas, nuestro modelo se encuentra dentro de la que dice modelo y es el archivo con terminación .txt.ply, este archivo puede ser abierto por el programa MeshLab.

Los procesos que se llevaron a cabo para el procesamiento de las imágenes durante la creación del modelo de la nube de puntos se describen a continuación [3, 4, 5].

SIFT (transformada de escala invariante)

SIFT es una transformada invariante a los cambios de escala fue desarrollada por David Lowe y presentado en [1]; debido a su alta eficiencia en la detección de objetos y su sencillez matemática se ha convertido en una transformada empleada en procesos de detección de objetos en movimiento y reconstrucción de objetos en 3D a partir de la obtención de múltiples imágenes sobre una misma escena [5].

Detección de extremos en escala-espacio

Como primera etapa del sistema es necesaria la detección de puntos que nos ayuden a localizar el objeto de interés y determinar su tamaño, esto se logra al realizar la correspondencia de los puntos o píxeles que forman el objeto y son extraídos de múltiples imágenes que corresponden a diferentes vistas del

objeto de interés. El tener múltiples imágenes del mismo objeto ayuda a realizar el cálculo de invariancia en la escala y su ubicación espacial. Los cálculos mencionados pueden ser vistos como relaciones geométricas espaciales que permitan reconstruir una estructura artificial a partir de múltiples puntos.

Ha sido demostrado por Koenderink (1984) y Linderberg (1994) que bajo una variedad de suposiciones razonables el único kernel escala-espacio posible es la función de Gauss. Por lo tanto el espacio - escala de una imagen es definido con una función que se produce por la convolución de la función de escala variante de Gauss con la imagen como entrada.

Localización de puntos de interés

En orden para detectar los locales máximos y mínimos de cada punto muestra es comparado con sus ocho vecinos de la imagen en la que se encuentra, y nueve vecinos de la imagen en la escala siguiente y anterior. Es seleccionado si el punto muestra es mayor que todos sus vecinos o menor a todos ellos. La dificultad del cálculo de esta prueba está dada bajo el hecho que la mayoría de los puntos muestra son eliminados en las primeras iteraciones.

Se utiliza la serie de Taylor en su forma de escala-espacio para que un punto de interés sea proyectado y se encuentren sus posiciones en las diferentes imágenes que representan las diferentes vistas del mismo objeto. Múltiples pixeles representaran un solo punto de interés a estos se les nombrara como sub-pixeles. Sin embargo, algunos sub-pixeles en alguna de las imágenes pueden pertenecer o no al punto de interés, si uno de los puntos extremos encontrados es menor a un cierto valor de umbral, ese punto se considerará como ruido y es descartado como parte del punto de interés.

Asignación de orientación

Como se ha mencionado anteriormente, múltiples imágenes equivalentes a múltiples vistas del mismo objeto, tienen que ser organizadas según el ángulo de la toma radial de la imagen, por lo tanto, en este punto del proceso, se asigna a los puntos de interés el ángulo que le corresponde según la toma radial de la imagen.

Para ello se toman los puntos vecinos en torno a cada punto de interés en función de la escala y se calcula la magnitud y dirección del gradiente. Entonces se hace un histograma de dichas direcciones ponderado por la magnitud del gradiente. El mayor pico en el histograma indica la orientación del punto de interés. Si existen otros picos por encima del 80% del más importante, se usan para crear otros puntos de interés en la misma posición y escala pero con diferente orientación.

Descriptor de punto de interés

Para reconocer un objeto hay que obtener los rasgos que lo describen. El cálculo de estos descriptores se realizan utilizando alguna característica de los diferentes pixeles que conforman el objeto de interés, una de estas características puede la orientación que guarda el pixel en relación a su vecindario de acción, es habitual considerar un vecindario o ventana de acción de 16x16 alrededor del pixel a evaluar. De cada uno de los pixeles y su vecindario

correspondiente se obtiene un histograma de orientación y cada elemento del histograma de un pixel corresponde a una subdivisión de su vecindario, para esta sub-división se sugiere una relación de 4x4. Cada histograma de cada pixel se concatena para formar el descriptor o la matriz que describe al objeto [4, 5].

Correspondencia de los puntos de interés

En esta parte del sistema se realiza la correspondencia de los múltiples puntos de las diferentes imágenes corresponden a uno solo sobre el objeto de interés. Esta correspondencia se lleva a cabo analizando los descriptores de una misma región pero en diferentes imágenes. Se comparan los puntos de mayor valor entre los descriptores de la misma región y se calcula su similitud o distancia. Si los valores máximos de los descriptores son similares indica que se trata de la misma región, de lo contrario se descarta su correspondencia. Para esto es necesario que el usuario determina un valor de umbral al cálculo de similitud que determine si son correspondientes o no [3, 4, 5].

En esta parte del sistema se realiza la correspondencia de los múltiples puntos de las diferentes imágenes corresponden a uno solo sobre el objeto de interés. Esta correspondencia se lleva a cabo analizando los descriptores de una misma región pero en diferentes imágenes. Se comparan los puntos de mayor valor entre los descriptores de la misma región y se calcula su similitud o distancia. Si los valores máximos de los descriptores son similares indica que se trata de la misma región, de lo contrario se descarta su correspondencia. Para esto es necesario que el usuario determina un valor de umbral al cálculo de similitud que determine si son correspondientes o no [3, 4, 5].

Los algoritmos de vista estéreo múltiple son usados para generar reconstrucción densa de objetos o escenas 3D. Esta técnica está basada en la medición de la función de consistencia, la cual nos dice si el modelo 3D es consistente con las imágenes de entrada. Generalmente esto es difícil debido al ruido en la imagen y los errores de calibración de la cámara.

Ejemplos de una función e consistencia son:

- Color: ¿Las cámaras ven el mismo color? Este acercamiento es válido solo para superficies Lambertiana y está basada en la medición de la varianza del color.
- Textura: ¿Es la textura alrededor del objeto la misma? Esta aproximación puede manejar superficies reflectantes, pero tiene problemas con objetos brillantes. Está basado en la medición de la correlación entre pixeles.

3.- Experimentación

En los experimentos realizados se usaron dos cámaras con las cuales se tomaron una serie de imágenes de varios objetos en múltiples vistas. La primera es una cámara digital SONY® DSC-S5000 y la segunda es una cámara digital Panasonic® DMC-FX9, ambas con una configuración similar y una resolución parecida.

Las imágenes se tomaron alterando la vista cada 30 grados de diferencia, al igual que el Angulo de inclinación de la cámara en 45 grados. Siendo reconstruidos satisfactoriamente los siguientes objetos: un hueso sobre un escritorio, un muñeco de peluche, y una figurilla de cerámica.

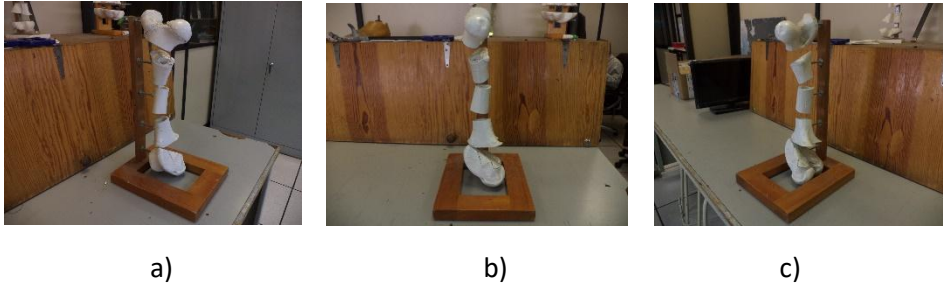


Figura 1 Fotografías muestra de un femur de bobino que se encontraba en el laboratorio. a) Hueso vista 45° izquierda, b) Hueso vista de frente, c) Hueso vista 45° derecha

Como se muestra en la Figura 1, se tomaron varias fotografías del objeto de interés tratando de mantener en el centro de la imagen dicho objeto. Siendo que se tomaron 12 fotografías para ver la respuesta que nos arrojaba nuestro programa, la reconstrucción fue exitosa como se muestra en la Figura 2.

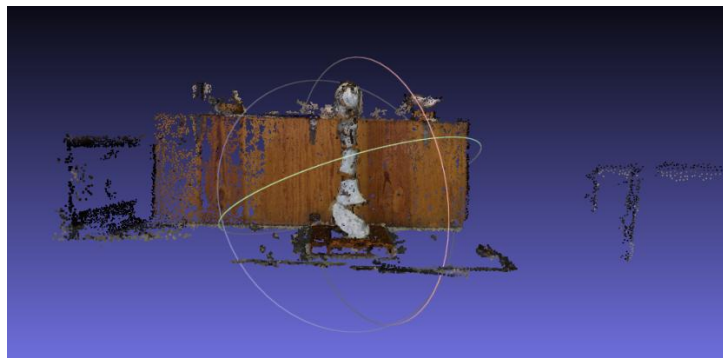


Figura 2 Nube de puntos que representa el hueso reconstruido

El software que se usó para poder observar el modelo de nube de puntos es MeshLab el cual nos permite manipular una amplia variedad de modelos en 3D. Como se puede observar la reconstrucción es exitosa, con la inclusión de que también tomo como puntos de interés el monitor a su izquierda y parte del marco de la puerta a su derecha.

Los huecos que se muestran en el hueso y en las superficies aledañas son debido a que en esas áreas no hay puntos clave suficientes para hacer "match". A continuación se muestran las fotos y los resultados de los otros objetos usando cada una de las diferentes cámaras disponibles.

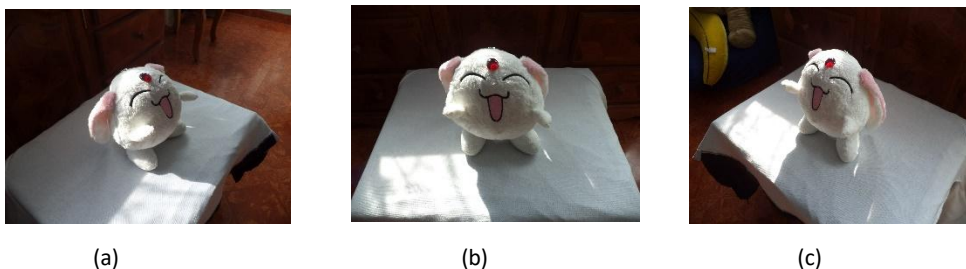


Figura 3 Muñeco de peluche tomado en diferentes ángulos: a) 45° derecha b) frente c) 45° izquierda

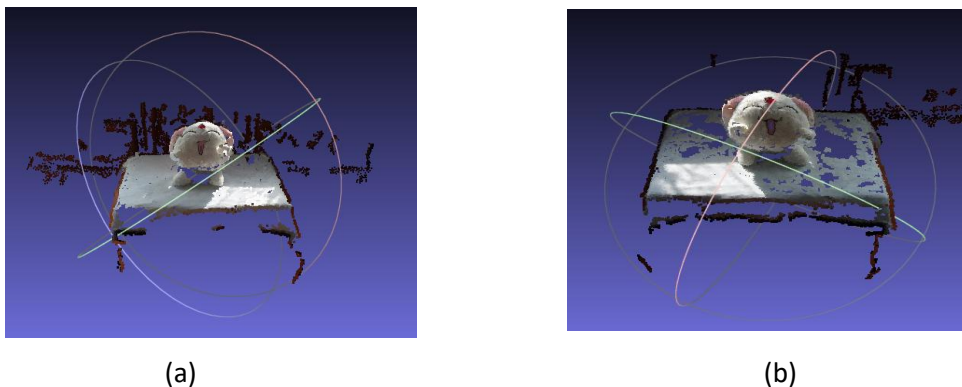


Figura 4 Reconstrucción del muñeco de peluche comparando cámaras. a) Cámara Sony® b) Cámara Panasonic®

Como se puede apreciar en la Figura 4, la cámara Sony® contiene un mayor detalle en la reconstrucción y se obtuvo una mejor reconstrucción de la superficie y de los alrededores de la figura gracias a un sensor Super HDA CCD que da una mejor captura de los detalles. Por otro lado la reconstrucción con cámara Panasonic® muestra una reconstrucción del objeto con ligeras fallas en la detección de la superficie y objetos aledaños.

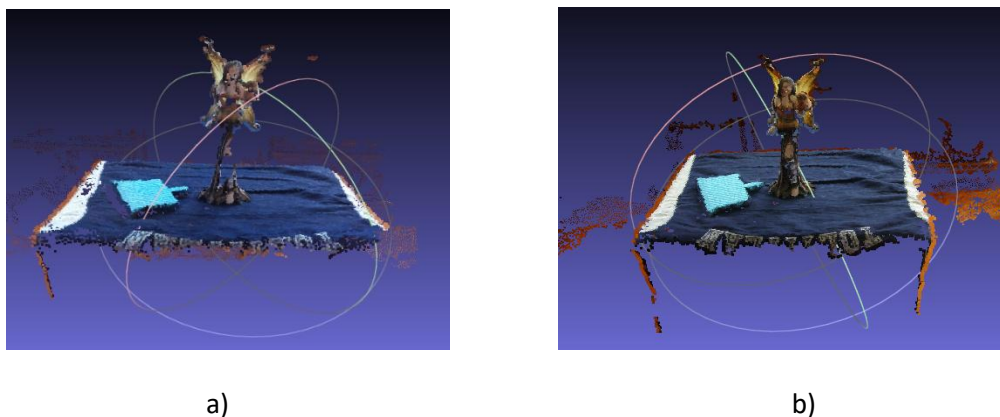


Figura 5 Figurilla de porcelana reconstruida para comparación de superficies: a) Cámara Sony® b) Cámara Panasonic®

En la Figura 5 se muestran los resultados de la reconstrucción de una superficie lisa que en comparación con la Figura 4 podemos notar menor cantidad de error en el modelo al igual que una pérdida en el

detalle. Esto debido a que la superficie del muñeco de peluche es rugosa y opaca en comparación con la superficie de la figurilla de porcelana que tiene una superficie lisa con cierto índice de refracción.

4.- Conclusiones

Los resultados muestran que con un mínimo de 12 fotos se puede tener una reconstrucción satisfactoria, con una gama de detalla suficiente para reconocer al objeto. Para una reconstrucción con mayor detalle no solo depende del número de fotos, también la resolución de la cámara que se usó para llevar a cabo la toma de fotografías de muestra tiene impacto en la reconstrucción. Siendo que para algunas superficies son detectadas con menor cantidad de huecos con cámaras de menor resolución que con una con mayor resolución, esto podría variar dependiendo del sensor de la cámara usada.

6.- Referencias

- [1] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 60, 2 (2004), pp. 91-110.
- [2] M. I.A. Lourakis and A.A. Argyros. The Design and Implementation of a Generic Sparse Bundle Adjustment Software Package Based on the Levenberg-Marquardt Algorithm. Tech. Rep. 340, Inst. Of Computer Science-FORTH, Heraklion, Crete, Greece. Available from <http://www.ics.forth.gr/~lourakis/sba>.
- [3] Yasutaka Furukawa, Carlos Hernández, "Multi-View Stereo: A Tutorial", Available from https://www.cse.wustl.edu/~furukawa/papers/fnt_mvs.pdf.
- [4] Shawn McCann, "3D Reconstruction from Multiple Images", available from http://cvgl.stanford.edu/teaching/cs231a_winter1415/prev/projects/CS231a-FinalReport-sgmccann.pdf
- [5] https://es.wikipedia.org/wiki/Scale-invariant_feature_transform