

LTI-UAV3D: una solución de alta calidad y bajo costo para la reconstrucción tridimensional del terreno empleando un μ -dron



LTI-UAV3D: a solution of high quality and low cost for terrain 3D reconstruction using a μ -uav



José-Gabriel Ramírez-Torres y Ander Larrañaga-Cepeda

CINVESTAV. Unidad Tamaulipas. Parque Científico y Tecnológico TECNOTAM. Carretera Cd. Victoria-Soto La Marina, Km. 5.5 – 87130 Cd. Victoria, Tamaulipas (México). Tfno: +52 834 107 0235.

DOI: <https://doi.org/10.6036/8174> | Recibido: 10/10/2016 • Evaluado: 14/10/2016 • Aceptado: 23/11/2016

ABSTRACT

- This article describes a simple and quite affordable tool called LTIUAV3D, for terrain three-dimensional reconstruction, using a low-cost unmanned aerial vehicle (UAV) and a personal computer. The 3D model does not require GPS information, because it is built exclusively from monocular images obtained by the onboard camera. To solve the scale factor of the 3D model, an artificial marker is used as a reference, so the obtained height map is metrically precise and correct. An important characteristic of LTIUAV3D is that the 3D model is built in real time, as the UAV flies over the terrain, with incremental updates every second. For the height map obtained, the estimation errors are less than 4%, so the 3D model could be used for path planning and autonomous navigation of UAVs. Also, the LTIUAV3D tool can get a much more precise and detailed 3D model, in an offline process, with height estimation errors less than 1%. This tool makes available, to a large number of users both individuals and academics, a robust foundation for the development of new applications of UAVs.
- **Keywords:** UAV, three-dimensional reconstruction, image processing, height map, real time.

RESUMEN

En este artículo se presenta una herramienta sencilla y bastante asequible, denominada *LTIUAV3D*, para la reconstrucción tridimensional del terreno, empleando un vehículo aéreo no tripulado (UAV) de bajo costo y una computadora personal. El modelo 3D no requiere información GPS, pues se construye a partir exclusivamente de las imágenes monoculares obtenidas por la cámara embarcada. Para resolver el factor de escala del modelo, se emplea un marcador artificial como referencia, de tal manera que el mapa de alturas es métricamente preciso y correcto. Una característica importante de *LTIUAV3D* es que el modelo 3D se construye en tiempo real, a medida que el UAV sobrevuela el terreno, con actualizaciones incrementales cada segundo.

En el mapa de alturas obtenido, los errores de estimación son inferiores a 4%, por lo que el modelo 3D podría emplearse para la planificación de trayectorias y la navegación autónoma del UAV. Asimismo, la herramienta *LTIUAV3D* puede obtener un modelo 3D mucho más preciso y detallado, en un proceso fuera de línea, con errores de estimación de altura menores a 1%.

Con esta herramienta se pone al alcance de un gran número de usuarios, tanto particulares como académicos, una base robusta para el desarrollo de nuevas aplicaciones de UAV.

Palabras clave: UAV, reconstrucción tridimensional, procesamiento de imágenes, mapa de alturas, tiempo real.

1. INTRODUCCIÓN

El principal objetivo de este artículo es presentar una herramienta sencilla y de bajo costo, denominada *LTIUAV3D*, que requiere únicamente de un μ UAV de bajo presupuesto equipado con una cámara y una computadora personal, para generar representaciones tridimensionales del terreno en tiempo real, de alta calidad y precisión, con información métricamente precisa de las elevaciones y las estructuras del medio ambiente, incluso cuando no se cuenta con señal GPS ni información georreferenciada.

Motivaciones

Un dron o vehículo aéreo no tripulado (UAV, en inglés) puede definirse como una máquina autopropulsada, que no lleva un operador humano a bordo, que emplea las fuerzas aerodinámicas para lograr la sustentación aérea y que puede volar de manera autónoma o ser controlado remotamente desde una estación en tierra [1]. Si bien cada país tiene su propia legislación para clasificar los UAV según sus características y capacidades, este trabajo está pensado para vehículos pequeños inferiores a 5 Kg, comúnmente denominados μ -UAV. Los UAV presentan ventajas como buen desempeño, acceso a zonas peligrosas y, principalmente, su capacidad de llevar sensores a puntos de vista privilegiados, su bajo peso y su bajo costo.

Es común describir la orografía de un terreno a través de un mapa de alturas, presentado al usuario como un mapa de colores. La generación de un mapa de alturas digital es un proceso que involucra la representación de la superficie del terreno a partir de una secuencia de lecturas (con telémetro láser [2] o cámaras estereoscópicas [3]), para establecer el modelo numérico conocido como Modelo Digital de Elevaciones (DEM), Modelo Digital del Terreno (DTM) o Modelo Digital de Superficie (DSM) [4,5]. Entre las aplicaciones de los DTM podemos mencionar la extracción de parámetros terrestres, la descripción de relieve cartográfico y la generación de mapas orto-rectificados, entre otras. Es posible encontrar en la literatura diferentes propuestas de generación de DTM a partir de imágenes con referencia GPS (Sistema de Posicionamiento Global, en inglés) obtenidas con un UAV [6-9].

Al día de hoy, los UAV son vehículos semiautónomos capaces de realizar tareas automáticas muy específicas, como la navegación basada en coordenadas georreferenciadas y despegue-aterrizaje automático [7]. Desarrollar algoritmos de navegación autónoma para UAV en medios ambientes de baja altitud requiere que el UAV tenga acceso a un mapa tridimensional del medio ambiente, con información precisa sobre distancias, alturas y ángulos de sus alrededores.

Esta representación del medio ambiente del UAV debe construirse en línea a partir de las lecturas obtenidas con los sensores embarcados; resolviendo un conjunto de problemas no triviales tales como la localización del UAV, las situaciones donde la señal GPS no está disponible y la evasión de obstáculos en tiempo real. En general, estos problemas son abordados empleando UAV con sofisticados sensores telemétricos, lo que requiere una inversión económica importante, quedando fuera del alcance de académicos y pequeñas y medianas empresas.

Enfoque propuesto: LTI-UAV3D

Por las razones expuestas anteriormente, una herramienta sencilla y asequible para generar un mapa tridimensional del terreno, sin realizar una importante inversión, empleando una computadora personal y un μ -UAV comercial, proporcionaría a particulares y académicos una base sólida para propósitos comerciales, como la reconstrucción de terrenos e inspección de estructuras, y para propósitos científicos, como el desarrollo de algoritmos de navegación autónoma.

La herramienta desarrollada, llamada LTIUAV3D, consiste en tres algoritmos computacionales que funcionan en paralelo: un algoritmo para el análisis digital de imágenes para generar una nube de puntos tridimensionales que describe el medio ambiente, tanto en línea como fuera de línea; un algoritmo para obtener el factor de escala real y conseguir un modelo métricamente correcto y preciso; y un algoritmo para la construcción del mapa de alturas.

Trabajos previos

Existen diferentes enfoques propuestos para la reconstrucción del medio ambiente a partir de imágenes obtenidas con un UAV [6,10]. Muchos autores sugieren el uso de sensores sofisticados (múltiples cámaras y datos GPS [10-13], cámaras estereoscópicas [3,9,14] o dispositivos LADAR [2]) para obtener la información de profundidad de las imágenes y facilitar la reconstrucción del medio ambiente.

Muchos algoritmos emplean descriptores de forma para realizar la reconstrucción. Se han propuesto distintos descriptores de forma para definir objetos en el espacio tridimensional [15-17], pero no son robustos a oclusiones y a la presencia de varios objetos similares en una sola escena. Lo mismo ocurre con los algoritmos clásicos de *Hashing* geométrico que se emplean comúnmente en la detección de objetos 3D [18].

En los enfoques de estructura a partir del movimiento (*Structure-from-Motion*, SfM) el resultado es una descripción de la escena en forma de una nube de puntos tridimensionales, geométricamente correcta pero sin información sobre la escala real, por lo que no es posible calcular las dimensiones absolutas de los objetos [19]. Como ejemplo podemos mencionar el algoritmo *Patch-based Multiview Stereo* (PMVS) [20] y el algoritmo *A Contrario Structure from Motion* (ACSfM) [21].

La escala real del modelo 3D puede introducirse de forma manual para construir el modelo en tiempo real, pero se requiere que la información de GPS esté disponible [2,6,13]. También es posible introducir una referencia artificial de dimensiones conocidas en la

escena para calcular la escala absoluta del modelo fuera de línea [22].

En todos estos algoritmos, el tiempo de procesamiento requerido para el emparejamiento de imágenes y cálculo de la pose de la cámara (posición y orientación de la cámara respecto al sistema coordinado de referencia global) es demasiado alto para aplicaciones de reconstrucción tridimensional en línea.

2. MATERIALES Y METODOS

Con el propósito de desarrollar una herramienta asequible para la mayoría de las instituciones académicas y usuarios particulares, para el presente trabajo nos hemos impuesto algunas restricciones materiales: la estación en tierra debe ser una computadora personal común con conectividad Wi-Fi, el μ -UAV debe ser un dispositivo comercial de bajo costo, sin localizador GPS y una cámara de baja resolución sin sistema de estabilización de imagen.

Para propósitos experimentales, empleamos el μ UAV *Parrot Ar.Drone* (Figura A en el material suplementario), por su bajo costo y el acceso al flujo de video y a los datos de navegación a través de una conexión Wi-Fi *ad-hoc*. Este μ -UAV es muy popular como plataforma experimental en diversos trabajos, por ejemplo la navegación en interiores utilizando los puntos de fuga en las imágenes [23], localización y estimación de trayectorias basadas en odometría [24-25] e interacción hombre-máquina apoyada en información visual [26].

La estación en tierra, con las rutinas de análisis de imágenes que componen LTIUAV3D, consiste en una computadora de escritorio *iMac* con procesador *Intel Core i5*, 4GB de RAM y conectividad Wi-Fi.

Descripción general de LTI-UAV3D

El problema bajo estudio puede plantearse de la manera siguiente:

Dado un μ -UAV que sobrevuela un terreno desconocido, con acceso en tiempo real al conjunto de imágenes tomadas por la cámara embarcada y a la información inercial del μ -UAV, generar un mapa de alturas del medio ambiente que pueda emplearse para estimar la posición, orientación y altitud del UAV, sin recurrir a información de GPS.

Así, mientras el μ -UAV sobrevuela un área desconocida, el sistema propuesto analiza el flujo de imágenes y construye un mapa de alturas del medio ambiente, sin utilizar información GPS, obteniendo una representación precisa y de alta calidad.

A conocimiento de los autores, hasta el día de hoy no se ha publicado un trabajo que utilice equipo de bajo costo para resolver el problema de construcción en tiempo real de un modelo 3D con información de elevación, empleando únicamente las imágenes monoculares de la cámara embarcada, incluyendo el cálculo del factor de escala.

Dentro de la herramienta LTIUAV3D se emplea una adaptación del algoritmo PMVS [20] para calcular rápidamente, a partir de la triangulación de las 3 últimas imágenes, una nube parcial de puntos 3D de la escena que será utilizada para actualizar el mapa global en tiempo real. El factor de escala del modelo se calcula con un marcador artificial (patrón cuadrado binario) que puede ser fácilmente detectado con un algoritmo de detección de marcadores [27], que también proporciona la pose de cámara. Estos marcadores pueden emplearse también para enviar órdenes de movimiento al μ -UAV, tales como "documentar esta región" o "esperar órdenes del usuario", por ejemplo.

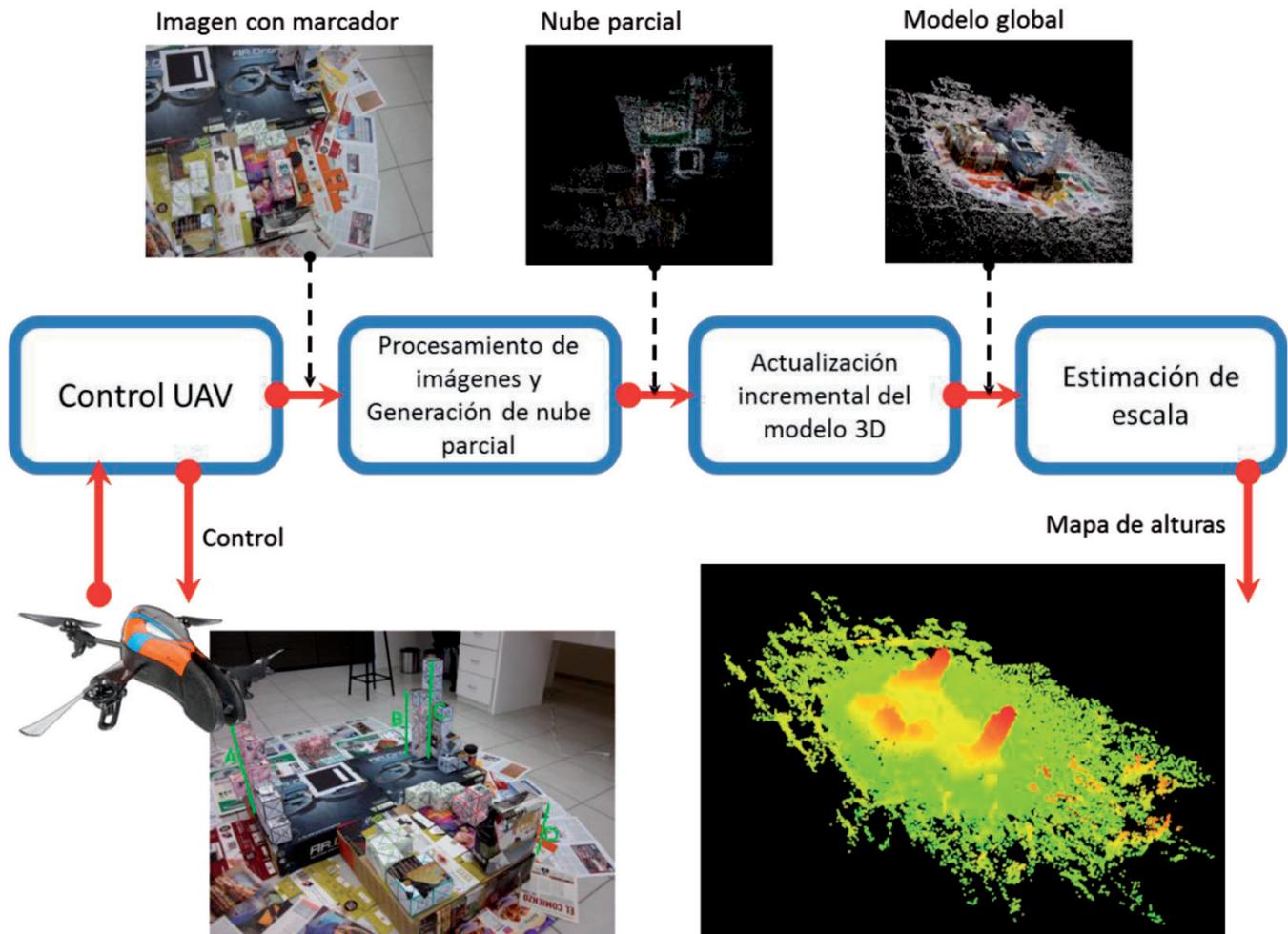


Figura 1: Esquema general de LTI-UAV3D

La herramienta LTIUAV3D está compuesta por 4 algoritmos organizados como un *pipeline* de módulos de procesamiento de información, como se muestra en la Figura 1: un módulo de control del μ -UAV, un módulo de generación de nubes parciales de puntos, un módulo de actualización incremental del modelo 3D y un módulo para la estimación de escala real del modelo, los cuales se detallan a continuación.

Módulo de Control UAV

El módulo de Control UAV establece el enlace de comunicación entre la estación en tierra y el UAV, que puede operarse de forma manual o seguir una trayectoria pre-programada, mientras analiza las imágenes buscando un marcador artificial que, de ser necesario, puede codificar información adicional [27].

El uso de marcadores artificiales provee tres ventajas muy útiles e importantes: primero, permite calcular la pose de la cámara en tiempo real, con poca sobrecarga computacional; segundo, el conocimiento *a priori* del tamaño del marcador permite calcular el factor de escala global del modelo; y tercero, el marcador permite establecer el nivel del suelo de la escena, es decir, el plano XY de referencia para las alturas estimadas.

La información adicional codificada en el marcador puede emplearse para establecer puntos de navegación en el terreno, interpretándola como órdenes sencillas, por ejemplo "Viaja Oeste" o "Espera aterrizaje", o incluso órdenes complejas como "Sobrevuela este punto" o "Localiza objetivo".

El módulo de Control UAV es un autómata, descrito en el Algoritmo 1, que controla el comportamiento actual del μ UAV, a través de órdenes dadas directamente por el operador o por la infor-

mación codificada en los marcadores artificiales. Si se detecta un marcador artificial, entonces la imagen actual y la matriz de pose de cámara son enviadas al módulo de Procesamiento de imágenes y Generación de nube parcial.

Algoritmo 1. Control UAV

```

Entrada: Comando Start y velocidades máximas ( $v_x, v_y, v_z$ )
Salida: Imagen  $I_n$  y matriz de pose de cámara  $C_n$ 

Despegue
while En_vuelo do
    Enviar órdenes de vuelo al UAV de acuerdo al estado actual
    Captura imagen  $I_n$ 
    if Marcador_navegación_detectado or Operador then
        Cambiar estado actual
    end if
    if Marcador_referencia_detectado then
        Calcula la matriz de pose de cámara  $C_n$  para la imagen  $I_n$ 
        Salida  $I_n$ 
        Salida  $C_n$ 
         $n = n + 1$ 
    end if
end while
Aterrizaje
    
```

Este módulo también mantiene un registro de la posición del μ -UAV, a través del cálculo de la odometría de los datos inerciales (Algoritmo 2).

Módulo de Procesamiento de imágenes y Generación de nube parcial

El módulo de Procesamiento de imágenes y Generación de nube parcial recibe las imágenes que contienen un marcador artificial, junto con la información de pose de cámara, y emplea una

Algoritmo 2. Odometría del UAV

Entrada: Datos inerciales: velocidades (v_x, v_y, v_z) y ángulos *yaw, pitch, roll*
Salida: Posición estimada $P = (x, y, z)$

```

 $t_{k-1} = t_k = \text{tiempo\_actual}$ 
 $P = (0, 0, 0)$ ; Posición inicial
while En_vuelo do
     $t_k = \text{tiempo\_actual}$ 
    Recibir datos inerciales (velocidades, ángulos)
     $dt = t_k - t_{k-1}$ 
     $M = [v_x dt, v_y dt, v_z dt]$ 
    Cálculo de matrices de rotación  $R_{yaw}, R_{pitch}, R_{roll}$ 
     $P = P + R_{yaw} R_{pitch} R_{roll} M$ ; Nueva posición estimada
     $t_{k-1} = t_k$ 
end while
    
```

adaptación del algoritmo PMVS [20] para generar una nube parcial de puntos tridimensionales a partir de bloques de 3 imágenes consecutivas, como se describe en el Algoritmo 3.

En nuestra adaptación del algoritmo PMVS, los puntos de interés (POI) son las esquinas y bordes dentro de la imagen, obtenidos con el algoritmo de Harris [28] y una diferencia de Gaussianas (DoG) [29] respectivamente, así como un algoritmo de "vecinos más cercanos" para encontrar los puntos consistentes con una transformación epipolar entre imágenes. Cada POI se modela como un "parche" (un punto 3D y un vector normal a la superficie). Los parches que aparecen en las tres imágenes se emplean para determinar por triangulación las coordenadas espaciales del parche.

Algoritmo 3. Cálculo de la nube parcial de puntos

Entrada: Imágenes I_1, I_2, I_3 y sus correspondientes matrices de pose de cámara C_1, C_2, C_3
Salida: Nube parcial de puntos *parcial*, con localización 3D (x, y, z) y color (R, G, B)

```

parcial =  $\emptyset$ ; Nube de puntos inicial
for cada imagen  $I_i$  do
    Detección del conjunto de POI  $I_i$ , usando detector de Harris
    Cálculo del conjunto de descriptores  $F_i$ 
end for
parcial =  $PMVS(F_1, F_2, F_3, C_1, C_2, C_3)$ 
    
```

Módulo de Actualización incremental del modelo 3D

El módulo de Actualización incremental del modelo 3D recibe una a una las nubes parciales generadas por el módulo anterior, y busca el mejor emparejamiento entre la nube parcial de puntos y el modelo 3D actual para fusionarlos. Las nubes parciales son libres de escala, cada una con su propia posición y orientación, independientes entre ellas. Además, existe un error no despreciable en la estimación de la pose de cámara de cada una de las imágenes que dificulta la fusión directa entre las nubes de puntos. Para lograr la correcta fusión de las nubes de puntos, empleamos un procedimiento de votación por *Transformada de Hough* para objetos 3D [30], sacando provecho de la superposición entre la nube parcial y el modelo completo, como se describe en el Algoritmo 4.

El algoritmo es el siguiente: se cuenta el número total de puntos en cada una de las nubes y se construye el vector normal para cada punto [31]. A continuación se efectúa un muestreo uniforme para dividir las nubes en pequeños cuboides, aproximados por sus respectivos centroides. Después, las nubes reducidas son descritas empleando descriptores tipo SHOT [32] y se encuentran los emparejamientos entre descriptores cuya distancia euclidiana es inferior a un valor umbral Hough predeterminado. Un valor pequeño del umbral Hough aumenta la precisión de la reconstrucción, pero reduce de forma importante la densidad de las nubes parciales. Para cada emparejamiento, con el algoritmo BOARD [33] se asigna un marco de referencia local (LRF) para obtener un descriptor invariante en traslación y rotación.

Como cada nube parcial está compuesta por parches encontrados en las tres últimas imágenes, es posible hacer la construcción incremental del modelo global al mismo tiempo que el μ -UAV

se desplaza en el medio ambiente, aunque con un impacto negativo sobre la calidad de la nube de puntos.

Algoritmo 4. Votación de Hough para emparejamiento 3D

Entrada: Nubes de puntos *escena* = $N_n(x, y, z, r, g, b)$ con n puntos y *parcial* = $O_m(x, y, z, r, g, b)$ con m puntos
Salida: Unión de nubes *escena* = $N_{n+m}(x, y, z, r, g, b)$

```

Cálculo normales_escena usando los 10 vecinos más cercanos para cada punto en escena
Cálculo normales_parcial usando los 10 vecinos más cercanos para cada punto en parcial
Cálculo keypoints_escena por muestreo uniforme en escena
Cálculo keypoints_parcial por muestreo uniforme en parcial
descriptores_escena =  $SHOT(\text{normales\_escena}, \text{keypoints\_escena})$ 
descriptores_parcial =  $SHOT(\text{normales\_parcial}, \text{keypoints\_parcial})$ 
 $LRF_{escena} = LRF(\text{escena}, \text{keypoints\_escena}, \text{descriptores\_escena})$ 
 $LRF_{parcial} = LRF(\text{parcial}, \text{keypoints\_parcial}, \text{descriptores\_parcial})$ 
Cálculo de la mejor transformación de emparejamiento  $M_{best} = HOUGH(LRF_{escena}, LRF_{parcial})$ 
escena =  $MERGE(\text{escena}, \text{parcial}, M_{best})$ 
    
```

Módulo de Estimación de escala

Finalmente, el módulo de Estimación de escala localiza de manera precisa el marcador dentro del modelo 3D y utiliza las referencias métricas del marcador para estimar el factor de escala real del modelo, por un proceso de votación de Hough entre la nube total de puntos y el marcador de referencia conocido. El tamaño relativo del marcador permite resolver el problema de certidumbre de escala y obtener un modelo 3D del medio ambiente métricamente correcto.

Reconstrucción fuera de línea

La herramienta *LTI-UAV3D* también es capaz de generar el modelo 3D sin emplear un marcador artificial de referencia, en un proceso fuera de línea basado en el algoritmo ACSfM [21] que consiste en el emparejamiento de imágenes con descriptores SIFT [34] y el cálculo de homografía entre imágenes para calcular las poses de la cámara. Finalmente, con todas las poses y las imágenes, el algoritmo PVMS genera la nube global de puntos tridimensionales. El proceso completo se describe en el Algoritmo 5.

Algoritmo 5. Reconstrucción fuera de línea

Entrada: Conjunto total de imágenes I_1, I_2, \dots, I_N
Salida: Nube de puntos *escena* = $N_n(x, y, z, r, g, b)$ con n puntos

```

Emparejamiento de imágenes con descriptores  $F_i$  tipo SIFT
Supresión de falsos positivos con algoritmo AC-RANSAC
Cálculo de poses de cámara  $C_1, C_2, \dots, C_N = ACSfM(F_1, F_2, \dots, F_N)$ 
escena =  $PMVS(F_1, F_2, \dots, F_N, C_1, C_2, \dots, C_N)$ 
    
```

3. RESULTADOS

En *LTI-UAV3D* la reconstrucción del medio ambiente es altamente dependiente del marcador artificial. Para validar el enfoque propuesto y estudiar el efecto del tamaño del marcador sobre la reconstrucción tridimensional y la estimación de alturas, se efectuaron diversas pruebas con diferentes tamaños de marcador. En esta sección se presentan los resultados para dos escenarios distintos con diferentes tamaños de marcador (Figura 2).

Para cada escenario se utilizaron cuatro alturas de referencia para calcular el error de estimación de alturas. En el *Escenario 1* se emplea un marcador de , mientras que en el *Escenario 2* se introdujo un marcador artificial de , para comparar la calidad entre ambas reconstrucciones.

En cada escenario, el μ -UAV realizó 31 experimentos (sobrevuelos) distintos sobre la maqueta a una altura aproximada de 2 m, mientras tomaba fotografías (Figura B en el material suplementario) con la cámara embarcada, resultando un conjunto de 72 imágenes de pixeles de resolución por cada experimento, para obtener el error promedio de estimación de alturas estadísticamente relevante.



Figura 2: Escenario 1 y Escenario 2, con alturas de referencia

Reconstrucción 3D en tiempo real

Las imágenes son procesadas por bloques de 3 imágenes consecutivas, logrando una reconstrucción incremental del medio ambiente mientras el robot sobrevuela el marcador artificial, a intervalos de un segundo por bloque, en promedio. Las Figuras C y D del material suplementario muestran el proceso de crecimiento de

la nube de puntos para el conjunto de imágenes del *Escenario 2* y la el relieve de la nube de puntos final, respectivamente.

Detección de marcadores y generación del mapa de alturas

Para el *Escenario 1*, con el marcador de , el mapa de alturas se muestra en la Figura 3, donde el color amarillo indica los puntos al nivel del marcador artificial,

el color rojo los puntos por encima y el color verde los puntos por debajo de dicho nivel. Se puede observar la contaminación de algunos puntos atípicos en la reconstrucción.

La Tabla I resume los resultados obtenidos para este escenario, con los valores promedios estimados para cada una de las alturas de referencia , para diferentes valores de umbral en el proceso de votación de Hough. La columna "Error relativo promedio (%)" muestra el error relativo promedio de la estimación de alturas para las 31 pruebas realizadas.

El mapa de alturas para el *Escenario 2*, con el marcador de , se muestra en la Figura 4. Es interesante notar que en este caso el número de puntos atípicos es significativamente menor al caso anterior, pues el tamaño del marcador en la imagen influye directamente en la precisión de la estimación

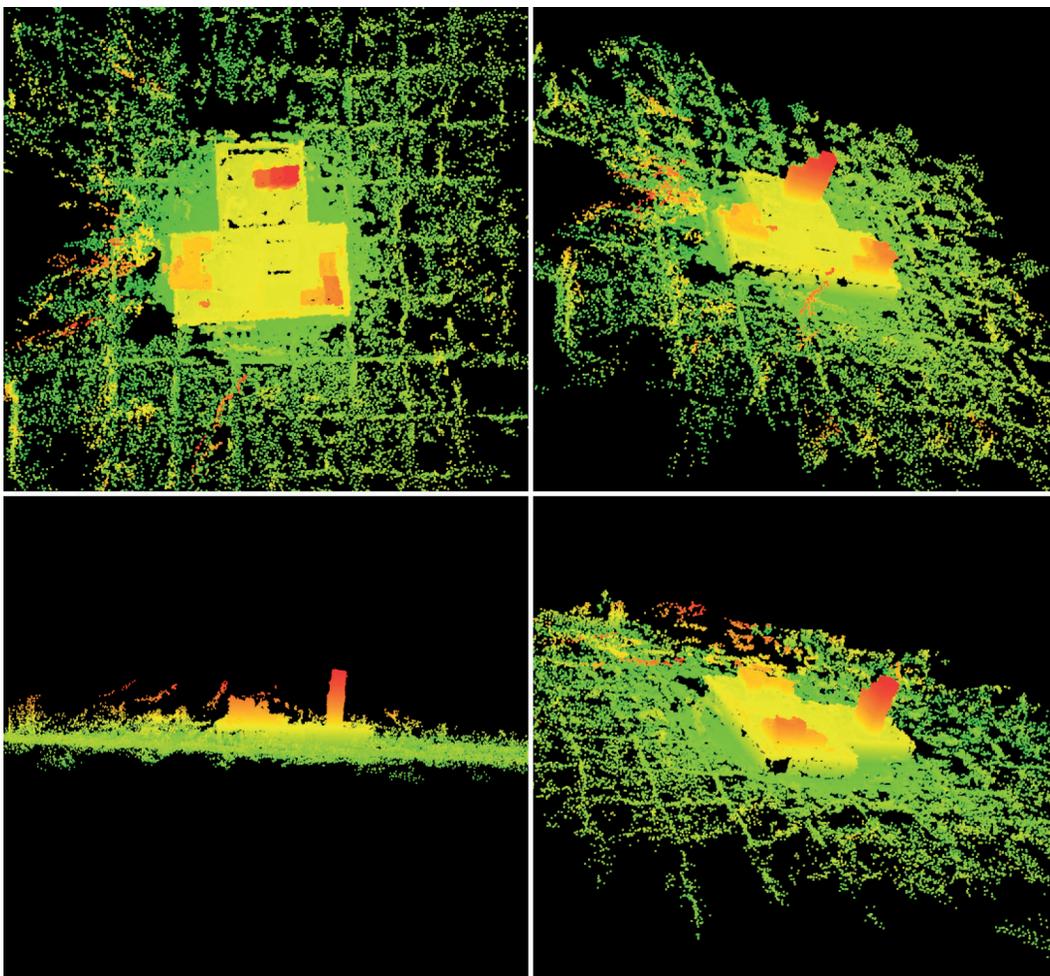


Figura 3: Mapa de alturas obtenido para el Escenario 1

Umbral Hough	Altura A (0.235 m)	Altura B (0.390 m)	Altura C (0.160 m)	Altura D (-0.14 m)	Error relativo promedio (%)
0.0100	0.2308	0.4046	0.1719	-0.1369	3.79
0.0150	0.2645	0.4477	0.1952	-0.1523	14.53
0.0200	0.3176	0.5474	0.2310	-0.1962	40.01

Tabla I: Resultados de estimación de alturas (metros) para el Escenario 1

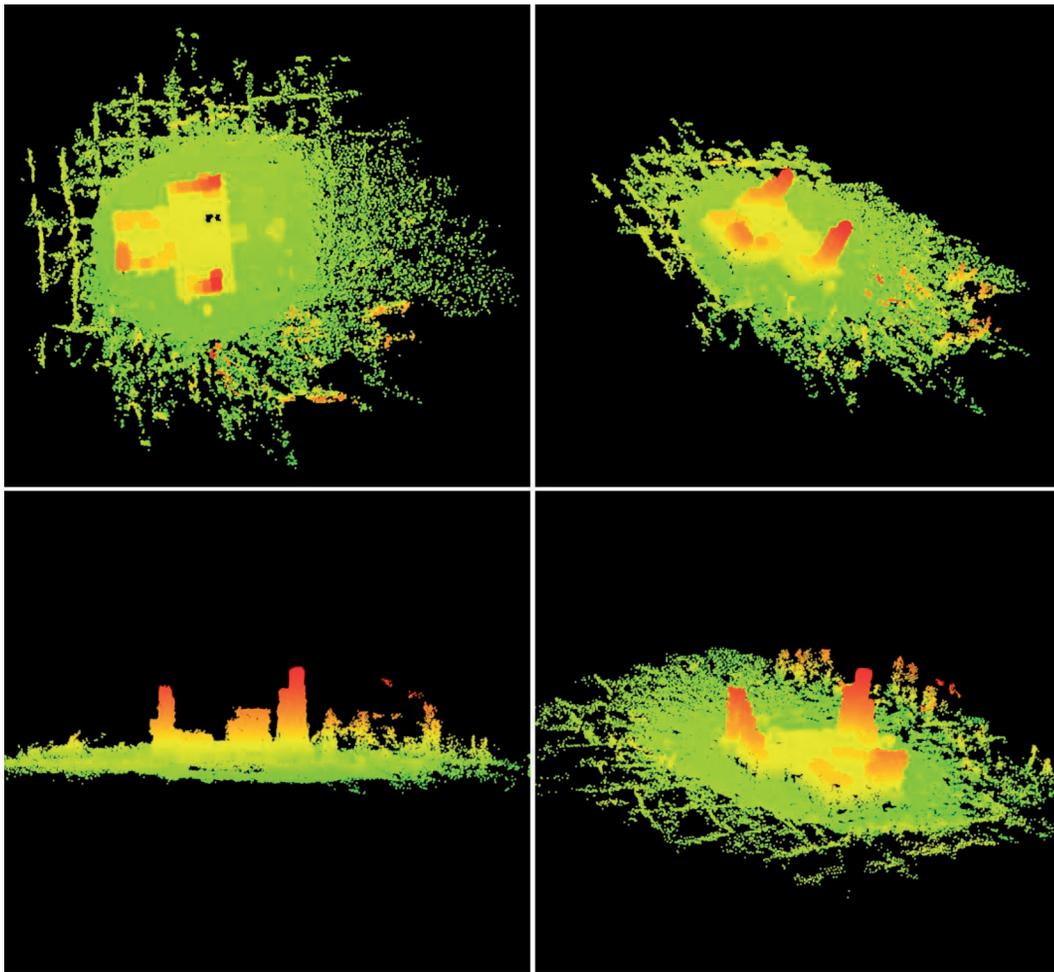


Figura 4: Mapa de alturas obtenido para el Escenario 2

tra el modelo 3D resultante del proceso de reconstrucción fuera de línea con las 72 imágenes del *Escenario 2*. A diferencia de las nubes de puntos obtenidos durante el proceso en línea, este modelo no contiene puntos atípicos y es más detallado y denso que los modelos 3D obtenidos al procesar las imágenes por bloques de 3 imágenes. Los errores de estimación son del orden de 1.5% de la altura máxima del medio ambiente.

Desempeño en escenarios realistas

Los resultados previos se obtuvieron en un medio ambiente controlado: las condiciones de luz y los objetos que se encuentran en la escena están condicionados para medir el desempeño de *LTI-UAV3D* respecto a los parámetros de la cámara (resolución) y del marcador (tamaño y ubicación).

Para evaluar *LTI-UAV3D* bajo condiciones más natu-

Umbral Hough	Altura A (0.390 m)	Altura B (0.395 m)	Altura C (0.550 m)	Altura D (0.240 m)	Error relativo promedio (%)
0.0100	0.4195	0.4098	0.5645	0.2471	4.22
0.0150	0.4261	0.4039	0.5707	0.2457	4.41
0.0200	0.4329	0.4143	0.5814	0.2525	6.70
0.0300	0.4398	0.4181	0.5777	0.2513	7.09
0.1000	0.5171	0.4985	0.7001	0.3071	28.51

Tabla II: Resultados de estimación de alturas (metros) para el Escenario 2

de la pose de la cámara, lo que se refleja en la precisión del proceso de triangulación de puntos 3D.

Los resultados obtenidos para el *Escenario 2* se resumen en la Tabla II y muestra la altura estimada promedio para cada altura de referencia, para distintos umbrales Hough, así como el error relativo promedio para las 31 pruebas realizadas.

Una observación importante es el efecto que el valor del umbral Hough tiene sobre la precisión del modelo 3D, la cual se reduce rápidamente a valores mayores de umbral. Este valor opera como una tolerancia durante el proceso de emparejamiento de imágenes: si el valor es muy pequeño, el número de emparejamientos válidos se reduce y la densidad de la nube de puntos se ve directamente afectada. Es necesario encontrar un compromiso entre densidad y precisión del modelo 3D deseado.

Reconstrucción 3D fuera de línea

Con el conjunto completo de imágenes es posible obtener una mejor reconstrucción tridimensional del escenario. La Figura 5 mues-

rales y realistas, se realizó un experimento adicional: se colocó un marcador de referencia en un ambiente de corredor, con una escalera de pasamanos tubulares, iluminada por luz solar a través de ventanales (Figura E en el material suplementario). El marcador se colocó en el suelo y se rodeó con papel decorado, para agregar textura y aumentar el número de puntos presentes en el modelo final.

El UAV sobrevoló el marcador tomando un conjunto de 25 imágenes, procesadas en tiempo real, para obtener la reconstrucción 3D y el mapa de alturas mostrados en la Figura F del material suplementario.

Se utilizaron cuatro elementos como alturas de referencia para evaluar el error promedio en la estimación de alturas. Los resultados obtenidos están resumidos en la Tabla III, con un error absoluto inferior a los 9 cm para una altura real de 2.3 m (errores relativos inferiores al 4%).

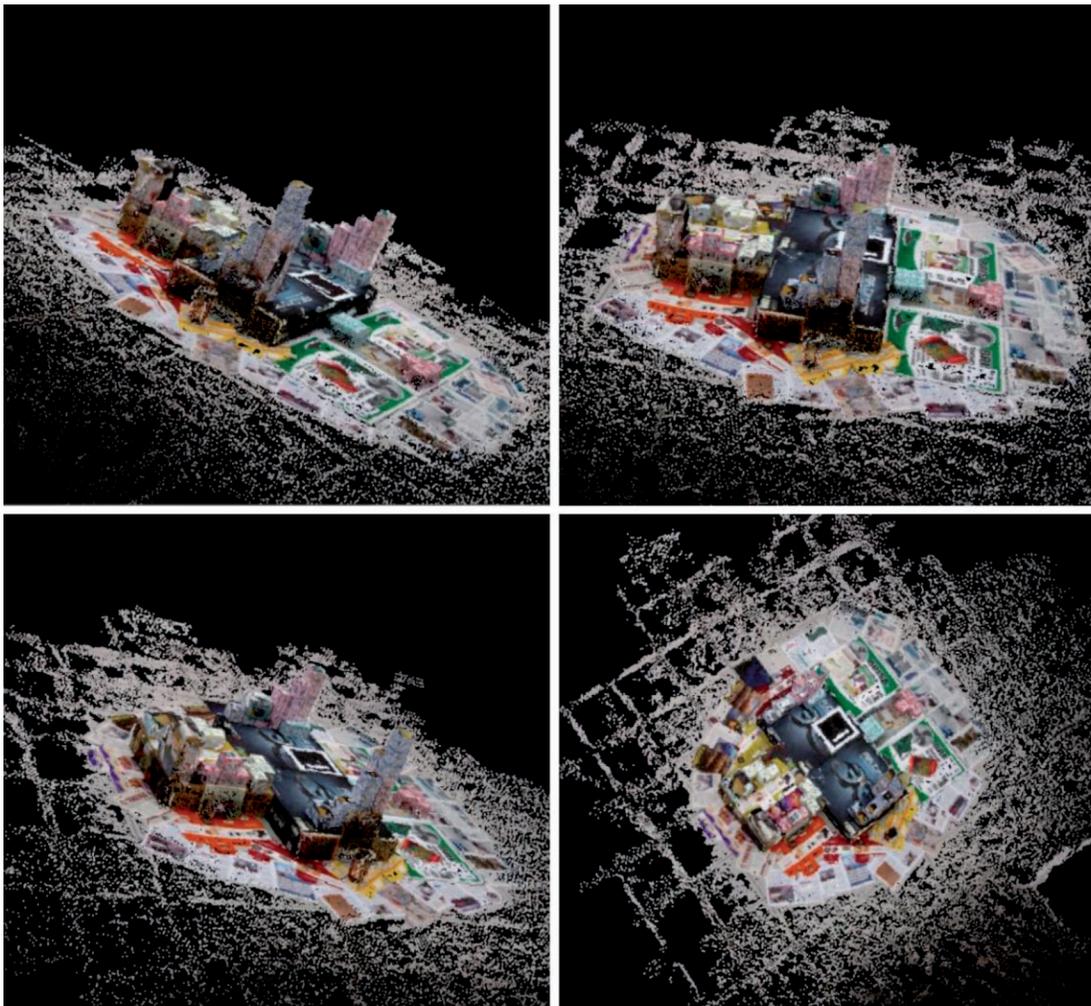


Figura 5: Modelo 3D construido fuera de línea

El tiempo de procesamiento de imágenes (1.05 bloques de 3 imágenes por segundo, en promedio) es compatible con los requerimientos de una aplicación en tiempo real. El error de estimación de alturas, con un umbral Hough de 0.01, es lo suficientemente pequeño (inferior a 4%) y el modelo 3D puede emplearse como una representación métricamente correcta del medio ambiente para propósitos de navegación autónoma del μ -UAV.

La herramienta LTI-UAV3D puede realizar también una reconstrucción fuera de línea, cuando el μ -UAV ha concluido el sobrevuelo del medio ambiente. Utilizando el marcador artificial para resolver la incertidumbre de escala, se obtiene un modelo 3D que es preciso, métricamente correcto y libre de puntos

Altura referencia	Altura real [m]	Altura estimada [m]	Error absoluto [m]	Error relativo [%]
A	0.920	0.8945	0.0255	2.77
B	0.530	0.5344	0.0044	0.83
C	-0.170	-0.1659	0.0041	2.41
D	-2.300	-2.3823	0.0823	3.58

Tabla III: Resultado de la estimación de alturas para el Escenario 3

4. DISCUSIÓN

A través de los experimentos realizados hemos encontrado que la calidad y la resolución de la imagen tienen una influencia significativa sobre la estimación de la pose de cámara y la detección del marcador de referencia. Por lo tanto, es importante hacer un énfasis especial en el desempeño de la cámara embarcada y en su adecuada calibración previa los sobrevuelos.

La herramienta LTI-UAV3D, empleando imágenes de píxeles, provee un mapa de alturas preciso, con tiempos de procesamiento adecuados utilizando una plataforma hardware muy asequible, que puede emplearse para el desarrollo de algoritmos de control autónomo. Como no utiliza información GPS, se pueden desarrollar aplicaciones para interiores de edificios y zonas donde el GPS no está disponible.

El tamaño del marcador artificial tiene un gran impacto en la calidad de la reconstrucción y en el número de puntos atípicos introducidos en el modelo. Asimismo, de los resultados reportados en las Tabla I y II, es posible inferir que, junto con el valor del umbral Hough, afecta también el proceso de estimación del factor de escala del modelo.

atípicos, donde las dimensiones, texturas y superficies del medio ambiente son fácilmente reconocibles.

Por los resultados obtenidos en el experimento en condiciones reales, podemos concluir que LTI-UAV3D cumple el objetivo propuesto, al realizar una reconstrucción 3D del medio ambiente, en tiempo real, precisa y detallada, para ser empleada en planificación de trayectorias y navegación autónoma del μ -UAV.

5. CONCLUSIONES

En este documento hemos presentado LTIUAV3D, una herramienta sencilla y de bajo costo, para la construcción de un modelo tridimensional del terreno en tiempo real, de alta calidad y precisión, a partir de imágenes sin información georreferenciada, empleando una plataforma de hardware asequible: un μ UAV comercial de bajo costo y una computadora personal.

Deseábamos que el hardware requerido en este desarrollo resultara tan asequible como fuera posible, para ofrecer a aca-

démicos y particulares una base sólida para abordar problemas más complejos, como la navegación reactiva autónoma, SLAM y recolección suficiente de datos.

Para este documento, realizamos diversos experimentos utilizando una computadora de escritorio y un μ UAV *Parrot AR.Drone 1*. Con imágenes de píxeles de resolución, la herramienta *LTIUAV3D* produce 1.05 incrementos por segundo al modelo 3D, suficiente para aplicaciones en tiempo real, dado que el μ UAV debe desplazarse para producir imágenes suficientemente diferentes, por lo que éstas llegan naturalmente a un ritmo menor. El modelo 3D es preciso y detallado (con errores inferiores a 4%).

Podemos concluir que *LTIUAV3D* provee un modelo 3D del medio ambiente equivalente, en términos de calidad visual, precisión métrica y tiempo de procesamiento, a los trabajos existentes en el estado del arte científico y tecnológico, empleando únicamente una plataforma hardware de bajo costo al alcance de un gran número de usuarios comerciales y académicos.

Actualmente estamos trabajando en una versión embarcada de *LTI-UAV3D*, para que el procesamiento se realice a bordo del μ UAV en una computadora de placa reducida (por ejemplo, una *Raspberry Pi*), evitando la transmisión de imágenes y aumentando la robustez del sistema en general.

Como trabajo futuro, consideramos extender *LTIUAV3D* a imágenes aéreas de ambientes exteriores, para estudiar el impacto que la altura de sobrevuelo del μ UAV, así como la luz y el contraste naturales, tienen sobre la generación y precisión del modelo. Asimismo, consideramos abordar los problemas de la reconstrucción tridimensional sin marcadores artificiales, la navegación autónoma y la reconstrucción 3D colaborativa.

BIBLIOGRAFÍA

- [1] Gertler J. "US Unmanned Aerial Systems". Library of Congress Washington DC Congressional Research Service. 2012.
- [2] Bybee TC, Budge SE. "Textured digital elevation model formation from low-cost UAV LADAR/digital image data". Proceedings of SPIE - the International Society for Optical Engineering, 2015, paper 131.
- [3] Haubeck K, Prinz T. "A UAV-based low-cost stereo camera system for archaeological surveys - experiences from Doliche (Turkey)". International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2013. 40(1W2) 195-200.
- [4] Sirmacek B, d'Angelo P, Krauss T, Reinartz P. "Enhancing Urban Digital Elevation Models Using Automated Computer Vision Techniques". International Conference on Pattern Recognition, 2010.
- [5] Weibel, R., M.Heller. "Digital Terrain Modelling". 1993.
- [6] Nex F, Remondino F. "UAV for 3D mapping applications: A review". Applied Geomatics, 2014, 6(1), 1-15.
- [7] Sanfourche M, et al. "Perception for UAV: Vision-Based Navigation and Environment Modeling". Journal AerospaceLab 2012, 4, 1-19.
- [8] Call, B. "Obstacle Avoidance For Unmanned Air Vehicles". Master's thesis, Brigham Young University, 2006.
- [9] Besnerais, G., Sanfourche, M., Champagnat, F. "Dense height map estimation from oblique aerial image sequences". Computer Vision and Image Understanding 2008, 109, 204-225.
- [10] Kršák B, Blišťan P, Paulíková A, Puškárová P, Kovanič L, Palková J, Zelížňaková V. "Use of low-cost UAV photogrammetry to analyze the accuracy of a digital elevation model in a case study". Measurement: Journal of the International Measurement Confederation, 2016, 91, 276-287.
- [11] Pollefeys, M., Nistér, D., et al. "Detailed Real-Time Urban 3D Reconstruction from Video". International Journal of Computer Vision 2008, 78, 143-167.
- [12] Barazzetti, L., Scaioni, M. "Orientation And 3D Modelling From Markerless Terrestrial Images: Combining Accuracy With Automation". The Photogrammetric Record 2013, 25, 2010.
- [13] Weiss, S., Achtelek, M., Kneip, L., Scaramuzza, D., Siegwart, R. "Intuitive 3D Maps for MAV Terrain Exploration and Obstacle Avoidance". Journal of Intelligent and Robotics Systems 2011, 61, 473-493.
- [14] Wen-Chung, C., Shu-An, L. "Real-time feature-based 3D map reconstruction for stereo visual guidance and control of mobile robots in indoor environments". 2004 IEEE International Conference on Systems, Man and Cybernetics, 2008, Vol. 6, pp. 5386 - 5391.
- [15] Mian, A., Bennamoun, M., Owens, R. "On the Repeatability and Quality of Keypoints for Local Feature-based 3D Object Retrieval from Cluttered

- Scenes". International Journal of Computer Vision 2010, 89, 348-361.
- [16] Chen, H., Bhanu, B. "3D free-form object recognition in range images using local surface patches". Pattern Recognition Letters 2007, 28, 1252-1262.
- [17] Zhong, Y. "Intrinsic shape signatures: A shape descriptor for 3D object recognition". Proc. 3DRR Workshop (in conj. with ICCV), 2009, pp. 689 - 696.
- [18] Lamdan, Y., Wolfson, H. "On the error analysis of geometric hashing". Proc. IEEE Conf. on Computer Vision, 1991, pp. 22-27.
- [19] Scaramuzza, D., Fraundorfer, F., Pollefeys, M., Siegwart, R. "Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints". 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 1413 - 1419.
- [20] Furukawa, Y., Ponce, J. "Accurate, dense, and robust multi-view stereopsis". Pattern Analysis and Machine Intelligence 2010, 32, 1362-1376.
- [21] Moulon, P., Monasse, P., Marlet, R. "Adaptive Structure from Motion with a Contrario Model Estimation". ACCV 2012, 2012, Vol. 7727, pp. 257-270.
- [22] Davison, A. "Real-Time Simultaneous Localization and Mapping with a Single Camera". 2003. Proceedings. Ninth IEEE International Conference on Computer Vision, 2003, Vol. 2, pp. 1403 - 141.
- [23] Bills, C., Chen, J., Saxena, A. "Autonomous MAV flight in indoor environments using single image perspective cues". International Conference on Robotics and Automation (ICRA), 2011.
- [24] Jimenez-Lugo, J., Zell, A. "Framework for Autonomous Onboard Navigation with the AR.Drone". International Conference on Unmanned Aircraft Systems 2013, pp. 575-583.
- [25] Engel, J., Sturmand, J., Cremers, D. "Camera-Based Navigation of a Low-Cost Quadcopter". Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International, 2012, Vol. 320.
- [26] Soto-Guerrero D, Ramírez-Torres JG. "A Human-Machine Interface with Unmanned Aerial Vehicles". Proceedings of the 10th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE '13), 2013, pp. 307-312.
- [27] Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F., Marín-Jiménez, M. "Automatic generation and detection of highly reliable fiducial markers under occlusion". Pattern Recognition 2014, 47, 2280 - 2292.
- [28] Harris, C., Stephens, M. "A combined corner and edge detector". Proceedings of the 4th Alvey Vision Conference, 1988, pp. 147-151.
- [29] Marr, D., Hildreth, E. "Theory of Edge Detection". Proceedings of the Royal Society of London, 1980, pp. 215-217.
- [30] Hough, P. "Method and means for recognizing complex patterns". US 3069654, 1962.
- [31] Khoshelham, K. "Extending generalized Hough transform to detect 3d objects in laser range data". Proc. ISPRS Workshop on Laser Scanning, 2007, pp. 206-210.
- [32] Tombari, F., S.Salti., DiStefano, L. "Unique signatures of histograms for local surface description". ECCV'10 Proceedings of the 11th European conference on computer vision conference on Computer vision, 2010, pp. 356-369.
- [33] Petrelli, A., Stefano, L.D. "On the repeatability of the local reference frame for partial shape matching". 13th International Conference on Computer Vision (ICCV), 2011, pp. 2244 - 2251.
- [34] Lowe, D. "Distinctive image features from scale-invariant key points". International Journal of Computer Vision 2004, 60, 91-110.

AGRADECIMIENTOS

Los autores desean agradecer el apoyo del fondo de becas de CONACYT de México durante la estancia de tesis de maestría en ciencias de Ander Larrañaga Cepeda, así como el apoyo del Laboratorio de Tecnologías de la Información del Cinvestav Tamaulipas por las facilidades prestadas para el desarrollo de este trabajo. Los autores expresan también su reconocimiento a los autores de las siguientes bibliotecas de software libre: OpenMVG, OpenCV, PMVS2 y Point Cloud Library.

MATERIAL SUPLEMENTARIO

http://www.revistadyna.com/documentos/pdfs/_adic/8174-1.pdf

