



**REPRESENTACIÓN AUTOMÁTICA EN 3D DE BAJO  
COSTE PARA INSPECCIÓN VISUAL  
DE COMPONENTES AERONÁUTICOS INSTALADOS  
EN ALA CON ACCESO LIMITADO**

**TESIS**

PARA OBTENER EL GRADO DE

**MAESTRO EN  
SISTEMAS INTELIGENTES MULTIMEDIA**

PRESENTA

**I.S.E. JOSÉ EDUARDO CORBAL MONTES  
ASESOR: DR. ISAÍAS EMMANUEL GARDUÑO OLVERA**

QUERÉTARO, QRO, OCTUBRE 2019

# CARTA DE LIBERACIÓN DEL ASESOR



GOBIERNO DE  
MÉXICO



CONACYT  
Consejo Nacional de Ciencia y Tecnología



Querétaro, Querétaro, 01 de agosto del 2019.

Mtro. Geovany González Carlos  
Coordinador Académico de Posgrado  
CIATEQ, A.C.

Los abajo firmantes, miembros del Comité Tutorial del ISE. José Eduardo Corbal Montes, una vez revisado su Proyecto Terminal de tesis, titulado "Software para sistema de micro visión para inspección de cámara de combustión On-Wing" **autorizo** que el citado trabajo sea presentado por el alumno para su revisión, con el fin de alcanzar el grado de **Maestro en Sistemas Inteligentes Multimedia**.

Sin otro particular por el momento, agradezco la atención prestada.

Firma

---

Dr. Isaías Emmanuel Garduño Olvera  
Asesor Académico

2019, Año del Caudillo del Sur, Emiliano Zapata

# CARTA DE LIBERACIÓN DEL REVISOR



Tepeapulco, Hidalgo, 1 de octubre de 2019.

Dra. María Guadalupe Navarro Rojero  
Directora de Posgrado  
CIATEQ. A.C.

Por medio de la presente me dirijo a usted en calidad de Revisora del proyecto terminal del alumno **JOSÉ EDUARDO CORBAL MONTES**, cuyo título es:

**"Representación automática en 3D de bajo coste para inspección visual de componentes aeronáuticos instalados en ala con acceso limitado"**

Después de haberlo leído, corregido e intercambiado información con el alumno, y realizo los cambios que le fueron sugeridos, puede ser autorizada su impresión, a fin de que se inicien los trámites correspondientes para su defensa.

Sin otro particular por el momento, y en espera de que mis sugerencias sean tomadas en cuenta en beneficio del estudiante y la Institución, agradezco la atención prestada.

Atentamente,

Dra. Carolina Reta Castro

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Cd. Sahagún, Hgo.

+52 (791) 913 7512

| 01 800 600 3798

| [mi@ciateq.mx](mailto:mi@ciateq.mx)

[www.ciateq.mx](http://www.ciateq.mx)

## RESUMEN

En el sector aeronáutico se realizan diferentes pruebas no destructivas con el objetivo de determinar el estado de los componentes de las aeronaves. Estas pruebas se realizan para determinar la cantidad de horas que puede seguir volando la aeronave con ese componente o si es necesario realizar algún mantenimiento inmediato. Debido a que se desea mantener la aeronave en operación la mayor cantidad de horas posibles, se desea realizar la mayor cantidad de pruebas sin desmontar la turbina del avión para evitar los tiempos que requieren el desmontaje de los mismos. Las pruebas de visión boroscópica, a pesar de que pueden realizarse en componentes con acceso limitado, son difíciles de ejecutar si dichos componentes se encuentran bloqueados, lo que dificulta aún más poder acceder a ellos sobre todo mientras la turbina se encuentra montada en el ala de la aeronave. Este estudio consiste en la optimización de pruebas de inspección a través de la captura, análisis y representación de un modelo 3D de la cámara de combustión de una aeronave. Se analizaron dos configuraciones posibles para la adquisición de imágenes: una sola cámara rotativa con captura de imagen panorámica y cuatro cámaras instaladas a 90° de separación. En el caso de la configuración con cuatro cámaras, se utilizaron scripts en Python utilizando librerías de procesamiento de imágenes para la creación de una imagen panorámica que representara el sistema. Los resultados obtenidos son bastante similares a la realidad en ambos casos y pueden ser utilizados por un inspector capacitado para la determinación del estado del componente. Sin embargo, la configuración con diferentes cámaras presentó una pérdida de 7% de los píxeles en comparación con la primera configuración, pues es necesario calibrar adecuadamente la distancia del desplazamiento del robot inspector para poder obtener la información de todo el escenario.

**Palabras clave:** Inspección visual, Representación 3D, Stitching.

## **ABSTRACT**

In the aeronautical sector, different non-destructive tests are carried out to determine the health status of aircraft components. These tests are performed to determine the number of hours the aircraft can continue flying with that component or if any immediate maintenance is necessary. Due to it is desired to keep the aircraft in operation as many hours as possible, it is wanted to carry out the biggest number of tests without disassembling the engine from the aircraft, as this task is time-consuming. Although the borescope inspection can reach zones with limited access, sometimes components are blocking some zones, making difficult the access if all parts are assembled. This study consists of the optimization of inspection tests through the capture, analysis, and representation of a 3D model of the combustion chamber of an aircraft. Two possible configurations for image acquisition were analyzed: a single rotating camera with panoramic image capture and four cameras installed at 90° of separation. In the case of the configuration with four cameras, Python scripts were used using image processing libraries for the creation of a panoramic image representing the system. The results obtained are quite similar to the reality in both cases and can be used by a trained inspector to determine the condition of the component. However, the configuration with different cameras showed a loss of 7% of the pixels compared to the first configuration. It is necessary to properly calibrate the distance to move the inspector robot to obtain the information of the whole scenario.

**Keywords:** Visual inspection, 3D representation, Stitching.

# ÍNDICE

1. INTRODUCCIÓN .....	1
1.1 ANTECEDENTES .....	1
1.2 DESCRIPCIÓN DEL PROBLEMA.....	1
1.3 JUSTIFICACIÓN .....	2
1.4 ALCANCES Y LIMITACIONES.....	3
1.5 OBJETIVOS.....	4
1.5.1 Objetivo general .....	5
1.5.2 Objetivos específicos.....	5
1.6 HIPÓTESIS .....	5
2. MARCO TEÓRICO .....	6
2.1 PRUEBAS NO DESTRUCTIVAS (PND) .....	6
2.1.1 Pruebas ultrasónicas.....	6
2.1.2 Termografía infrarroja .....	7
2.1.3 Pruebas basadas en vibración .....	8
2.1.4 Inspección visual.....	9
2.1.5 Robots para automatización de pruebas no destructivas.....	10
2.2 SISTEMAS DE INSPECCIÓN .....	11
2.2.1 Inspecciones visuales en ingeniería civil.....	11
2.2.2 Inspecciones de turbinas aeronáuticas .....	13
2.2.3 Mantenimiento de motores en aeronaves .....	14
2.2.4 Sistema de pronóstico y salud del motor .....	16
2.2.5 Tendencias en inspección de componentes aeronáutico.....	20
2.3 FOTOGRAMETRÍA Y RECONSTRUCCIÓN EN 3D.....	22
2.3.1 Triangulación pasiva.....	22
2.3.2 Agrupación de imágenes .....	23
2.3.3 Reconstrucción 3D con imágenes 2D .....	24
2.3.4 Escaneo láser.....	27
2.4 SOFTWARE .....	28
2.4.1 Reconstrucción en 3D.....	29
2.4.2 Modelado 3D.....	29
2.4.3 Visualización de modelos y motores de contenidos.....	30
3. PROCEDIMIENTO DE INVESTIGACIÓN .....	32
3.1 REPRESENTACIÓN DE ESCENARIO DE BAJO COSTO .....	32

3.1.1	Adquisición de imágenes .....	33
3.1.2	Definición de malla para utilizar en inspección boroscópica .....	35
3.1.3	Texturización de la malla .....	35
3.2	REPRESENTACIÓN EN 3D DEL ESCENARIO.....	36
3.2.1	Malla cilíndrica .....	36
3.2.2	Script en Blender .....	37
3.3	AGRUPACIÓN DE IMÁGENES .....	44
3.4	EJECUCIÓN DE PRUEBAS .....	45
4.	RESULTADOS .....	47
4.1	RESULTADOS CON FOTOS PANORÁMICAS.....	47
4.2	RESULTADO CON MÚLTIPLES FOTOS.....	50
	CONCLUSIONES.....	60
	APORTACIÓN DE LA TESIS.....	63
	RECOMENDACIONES.....	64
	REFERENCIAS BIBLIOGRÁFICAS .....	66
	ANEXOS.....	71

## ÍNDICE DE FIGURAS

Figura 1. Soft Robot creado por científicos de Harvard diseñado de material flexible que le permite deslizarse por áreas inaccesibles <sup>[21]</sup> . .....	2
Figura 2. Partes que conforman el desarrollo de un robot inspector.....	4
Figura 3. Termograma Infrarrojo realizado a la flecha de una turbina representando sus partes de acuerdo con la temperatura detectada en los mismos <sup>[48]</sup> .....	7
Figura 4. Posicionamiento del CHEOPS en una mesa de agitación para realizar pruebas de estructurales del satélite <sup>[41]</sup> . .....	8
Figura 5. Equipo para inspección boroscópica con pantalla para visualización <sup>[8]</sup> . .....	9
Figura 6. Vehículo autónomo sin piloto con cámara para inspección de estructuras <sup>[10]</sup> . .. .....	11
Figura 7. Sistema de bajo costo para inspección visual eficiente de túneles <sup>[45]</sup> .....	12
Figura 8. Ganancia neta de las aerolíneas de 2015 al 2018 <sup>[4]</sup> . .....	15
Figura 9. Principales beneficios de los sistemas de pronósticos <sup>[6]</sup> .....	16
Figura 10. Identificación de falla con límite de degradación de componente <sup>[6]</sup> . .....	17
Figura 11. Mapa visual sobre las partes que comprende la cámara de combustión <sup>[33]</sup> . ..	17
Figura 12. Cámara de combustión de un motor CFM56. Generalmente la inspección boroscópica se realiza recorriendo el Inner Liner <sup>[33]</sup> . .....	19
Figura 13. Grietas observadas dentro de una cámara de combustión <sup>[49]</sup> .....	20
Figura 14. El robot 'cucaracha' de 10 mm de Rolls Royce tiene por objetivo simplificar actividades de inspección y reparación en componentes aeronáuticos <sup>[20]</sup> .....	20
Figura 15. Robot de desplazamiento por principio de succión utilizado para análisis y determinación de curvaturas en estructuras aeronáuticas [47]. .....	21
Figura 16. El principio de triangulación pasiva se basa en la comparación de puntos de un objeto visto desde dos perspectivas diferentes <sup>[17]</sup> . .....	23
Figura 17. El stitching de imágenes (c) se logra tras la sobre posición de aquellas regiones comunes entre diferentes imágenes (estrella en a y b). .....	24
Figura 18. Configuraciones ofrecidas en Fiji para realizar el Stitching de múltiples imágenes <sup>[37]</sup> . .....	24
Figura 19. Pasos por seguir para una reconstrucción en 3D con imágenes 2D <sup>[29]</sup> .....	25
Figura 20. Entrada a cripta bizantina <sup>[27]</sup> .....	25
Figura 21. Malla de la entrada a cripta bizantina <sup>[27]</sup> . .....	26

Figura 22. La figura de la izquierda es la malla creada. La figura del centro es el refinamiento total de la malla. La figura de la derecha muestra un refinamiento local en la parte inferior central <sup>[31]</sup> .....	26
Figura 23. Ejemplo de texturización de una malla <sup>[27]</sup> .....	27
Figura 24. Principio de adquisición de información con un rayo láser <sup>[28]</sup> .....	28
Figura 25. A la izquierda el modelo 3D obtenido por escaneo láser. A la derecha el modelo 3D con la textura obtenida por fotogrametría <sup>[27]</sup> . ....	28
Figura 26.a)Configuración con una sola cámara con un eje rotativo para adquisición de imágenes panorámicas.b)Configuración de 4 cámaras separadas entre si 90°.c)Integración de anillo de iluminación a la configuración con 4 cámaras. ....	34
Figura 27. Malla cilíndrica generada en Blender. ....	35
Figura 28. Función 'Unwrap' en Blender de una malla cilíndrica.....	36
Figura 29. Una sola malla cilíndrica con n cortes para representar cada sección del escenario.....	37
Figura 30. Múltiples mallas que representan una sección del escenario fragmentado. .	37
Figura 31. Diagrama de flujo de alto nivel que muestra el funcionamiento del script de Blender.....	38
Figura 32. Dentro de la configuración inicial se encuentra el folder donde se localizan las imágenes. De igual manera, se define la posición donde se encontrará la primera malla. ....	39
Figura 33. Código para la creación y configuración de la malla cilíndrica.....	40
Figura 34. Visualización de la primera malla en Blender.....	40
Figura 35. Código de preparación de la malla para realizar la texturización. ....	41
Figura 36. A la izquierda la configuración flat de la malla. A la derecha la configuración smooth. ....	41
Figura 37. Código para la adición de textura a la malla. ....	42
Figura 38. A la izquierda mapa de texturización antes de realizar transformaciones. A la derecha, mapa de texturización después de realizar las transformaciones.....	43
Figura 39. Ejemplo de adición de nueva malla (sección).....	43
Figura 40. Imagen del interior de la cámara de combustión. La figura de la izquierda muestra la pared de inyectores y los orificios de presión. La figura de la derecha muestra la rejilla de ventilación entre las dos paredes laterales con orificios de presión <sup>[51]</sup> . ....	45

Figura 41. Imagen panorámica adquirida dentro de un elevador. ....	47
Figura 42. a) Modelo generado en Blender conformado por 5 mallas cilíndricas. b) Vista frontal del modelo en Sketchfab. c) Vista interior del modelo 3D en Sketchfab. ....	48
Figura 43. Imagen panorámica del interior de un ducto de ventilación.....	49
Figura 44. a) Representación del modelo generado. b) Fotografía interna del lado superior del modelo. c) Fotografía interna del lado inferior derecho del modelo.....	50
Figura 45. Imágenes capturadas a distintos ángulos dentro de un ducto de aire.....	51
Figura 46. Mensaje de error de la función create_Stitcher de Python.....	51
Figura 47. Imágenes del ducto de aire producido con Fiji.....	52
Figura 48. Dimensiones del horno inspeccionado. ....	52
Figura 49. Imágenes capturadas a distintos ángulos dentro del horno de una estufa. ..	53
Figura 50. Área analizada para determinar diferencias entre el perímetro real y el perímetro capturado en las imágenes.....	54
Figura 51. Resultados obtenidos con diferente software de la creación de una imagen panorámica a partir de 4 imágenes. a) Imagen panorámica capturada, b) Resultado obtenido por el script de OpenCV con la función create_stitcher., c) Resultados obtenidos por Fiji con el plug-In 'Stitching'. ....	55
Figura 52. Visualización del modelo creado a partir de imágenes: a) Vista exterior del modelo, ..b) Vista desde el interior del modelo, c) Vista de la parte superior del modelo .....	56
Figura 53. Visualización del modelo creado a partir de imágenes: a) Vista exterior del modelo, ..b) Vista desde el interior del modelo, c) Vista de la parte superior del modelo .....	57
Figura 54. Imágenes de inspección boroscópica de una cámara de combustión. a) Imagen lado izquierdo, b) Imagen lado derecho, c) Stitching de las imágenes. <sup>[3]</sup> .....	58

## ÍNDICE DE TABLAS

Tabla 1. Análisis comparativo de sistemas de inspección en la industria aeronáutica. .....	10
Tabla 2. Comparativa en tiempo de procesamiento y calidad del modelo 3D de diferentes programas para reconstrucción en 3D.....	29

## GLOSARIO

API	Conjunto de funciones o procedimientos que cumplen una o varias funciones y que pueden ser utilizadas por otro software
EASA	European Aviation Safety Agency
FAA	Federal Aviation Administration
Inner Linner	Cavidad o ducto localizado entre las paredes que conforman la cámara de combustión
On-Wing	Montado en el ala
Outer Linner	Hace referencia a la cara exterior de las paredes que conforman la cámara de combustión
PHM	Prognostics Health Management (Pronósticos para la gestión de salud)
PND	Pruebas no destructivas
TRG	Topological Relationship Graph (Gráfico de Relación Topológica)
UAV	Unmanned Autonomous Vehicle (Vehículo Autónomo sin Piloto)

# **1. INTRODUCCIÓN**

## **1.1 ANTECEDENTES**

Las actividades de mantenimiento en turbinas aeroespaciales requieren de una mayor eficiencia y rapidez. Por esta razón, el análisis y estudio de nuevas técnicas sobre la ejecución de estas actividades se ha vuelto parte fundamental en este sector. Al igual que en otros productos, el mantenimiento que se realiza a las turbinas puede ser preventivo o correctivo. A diferencia del mantenimiento correctivo, donde se identifica el problema a través de cierta información de la turbina y con el uso de manuales se aplica la acción correctiva, el mantenimiento preventivo requiere en ocasiones la inspección de ciertas partes de la turbina que no necesariamente están dañados<sup>[22]</sup>.

Para la ejecución de ciertas actividades de mantenimiento, en las aeronaves más recientes, los operadores cuentan con manuales y software especializado<sup>[22]</sup> para la realización de pruebas de verificación de algunas partes esenciales como: energización de válvulas, despliegue de compuertas para direccionamiento de aire, inyección de combustible, entre otros; a través de los cuales son capaces de determinar la mejor acción a tomar. Sin embargo, no todas las aeronaves cuentan con este tipo de sistemas de monitoreo e inclusive los que cuentan con ellos aún presentan ciertos problemas en partes mecánicas relacionados con el deterioro de estas piezas, para lo cual se suelen realizar una serie de pruebas no destructivas con el objetivo de encontrar si el componente presenta algún tipo de avería o no.

## **1.2 DESCRIPCIÓN DEL PROBLEMA**

El software del controlador del motor de la turbina ejecuta actividades de monitoreo constantemente para determinar su estado de salud. A través de la detección de comportamientos no esperados, como por ejemplo altas temperaturas o sobre presurización en ciertas regiones de la turbina, es posible reportar fallas para la ejecución de actividades de mantenimiento. Algunas fallas reportadas por el sistema de monitoreo requieren ser verificadas antes de continuar con la operación de la aeronave. Para algunas piezas la única forma de realizar dicha verificación es a través de pruebas no destructivas. Con el objetivo de evitar tiempos adicionales innecesarios, se desea ejecutar estas pruebas sin desensamblar el componente, procediendo únicamente al

desensamble una vez confirmado la necesidad de realizar alguna reparación o mantenimiento. La inspección boroscópica suele ser una de las primeras alternativas por ser de las más sencillas. A pesar de esto, existen ciertas complicaciones:

- a. Difícil acceso a ciertas partes cuando se encuentran ensambladas. La cámara de combustión de una turbina es un ejemplo de ellos, la cual cuenta con un acceso de tan solo 6 mm cuando se encuentra ensamblada en el ala de la aeronave.
- b. Requieren la presencia de un especialista en dicho componente para su correcta interpretación. En algunos aeropuertos se suele trasladar la turbina o aeronave a talleres especializados para la ejecución de estas pruebas, generando tiempos adicionales.

Debido a esto se ha considerado aquellos robots conocidos como suaves o soft-robots, como el mostrado en la Figura 1. Este tipo de robots se caracterizan por los materiales con los cuales son manufacturados. Mientras un robot convencional es de materiales rígidos (por ejemplo, plásticos duros, aleaciones, entre otros), un soft-robot se caracteriza por conformarse de materiales similares a aquellos de los que se compone un ser vivo (por ejemplo, piel y tejidos). Entre sus principales ventajas se encuentra su potencial para mejorar la interacción robot–persona, así como sus grandes capacidades de movilidad, a través de las cuales el robot es capaz de deformar su estructura para superar los obstáculos que se le presenten, algo que los robots de materiales rígidos no son capaces de realizar<sup>[21]</sup>.



*Figura 1. Soft Robot creado por científicos de Harvard diseñado de material flexible que le permite deslizarse por áreas inaccesibles<sup>[21]</sup>.*

### **1.3 JUSTIFICACIÓN**

Las capacidades de movilidad de un soft-robot resultan de gran ayuda para llegar a zonas con acceso limitado, convirtiéndolo en una alternativa bastante interesante para la ejecución en tareas de inspección. Para que la aplicación se pueda llevar a cabo de manera automática, es necesario la creación de un sistema embebido que sea capaz de controlar dicho robot. Este proyecto se enfoca en el proceso de inspección *On-wing* automático de la cámara de combustión enfocado a la reducción de tiempo requerido para realizar la inspección manualmente, evitando consiguos tiempos operativos como el montaje y desmontaje de la turbina. El software del sistema embebido se compone de diferentes módulos para la ejecución de cada una de sus funciones:

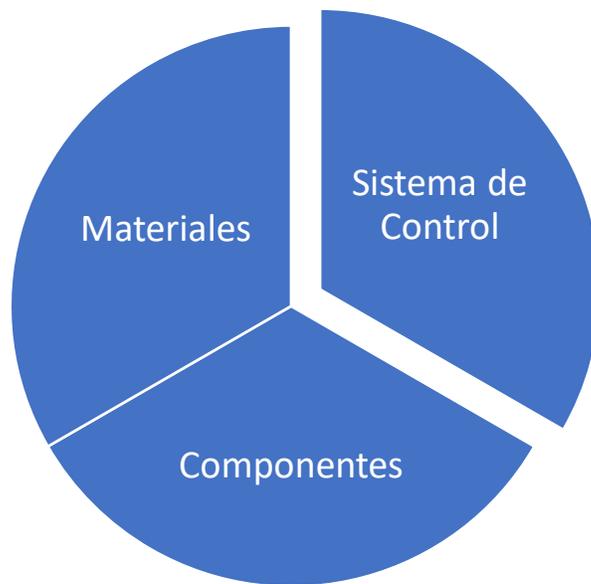
- a. Módulo de desplazamiento. Se encarga de controlar el movimiento del robot a lo largo del sistema a inspeccionar.
- b. Módulo de adquisición de imágenes. Controla la cámara para la captura de fotografías que mostrarán el estado actual del escenario a inspeccionar y se encarga de la transmisión de las mismas para su futura visualización por el inspector.
- c. Módulo de visualización. Toma las imágenes capturadas y crea una representación del escenario que pueda ser utilizada por el inspector para realizar un recorrido virtual del escenario, sin la necesidad de haber estado presente durante la inspección.

## 1.4 ALCANCES Y LIMITACIONES

Es posible describir el proyecto completo del soft-robot inspector en tres grandes categorías (como se muestra en la Figura 2):

1. **Materiales.** Esto hace referencia al estudio y análisis sobre los diferentes materiales con los que se puede construir. Estos materiales determinarán características como la estructura y forma del robot, así como sus atributos de movilidad.
2. **Componentes.** Son aquellos objetos físicos que integran al robot para la realización de funciones específicas. Aquí podremos encontrar: cámaras para la adquisición de imágenes, cableado para alimentación de energía y transmisión de información y algunos otros componentes que sean requeridos para la movilidad y soporte de la estructura del robot.
3. **Sistema de control.** Se encarga de controlar el comportamiento del robot. Le indica cuando desplazarse y hacia donde; también se encarga de controlar las

cámaras para la adquisición y procesamiento de imágenes y envía la información para ser analizada.



*Figura 2. Partes que conforman el desarrollo de un robot inspector.*

La categoría relacionada a materiales y componentes no serán abarcadas en este proyecto debido a que se encuentran siendo analizadas de manera separada. En cuanto a la categoría del sistema de control, el presente estudio se enfoca en la adquisición y visualización del escenario inspeccionado, dejando a un inspector la tarea de la determinación del estado actual del objeto inspeccionado. Las actividades de movilidad tampoco serán abarcadas en este estudio por la dependencia que tienen con los materiales y componentes que pudieran llegar a conformar el robot inspector. Considerando estas limitaciones, es muy importante mencionar que esta tesis se centró en el diseño y desarrollo del sistema de control de un robot inspector de manera modular, para que en un futuro sea posible la adición de las funcionalidades no abarcadas en este trabajo, tras la construcción de un prototipo. A pesar de lo anterior y debido a la relación directa con la adquisición de imágenes, en este trabajo se analizarán y recomendarán posibles cámaras a utilizar por este robot.

## **1.5 OBJETIVOS**

### **1.5.1 Objetivo general**

Proveer un software para crear una representación automática en 3D de bajo coste, que muestre el estado actual del interior de componentes aeronáuticos con acceso limitado cuando se encuentran instalados en el ala de la aeronave, a través de la adquisición de imágenes obtenidas por un robot, para que un inspector pueda determinar la operabilidad del mismo.

### **1.5.2 Objetivos específicos**

- a. Representar el escenario del objeto inspeccionado en un sistema 3D que pueda ser visualizado por el operador para su valoración.
- b. Proveer una solución de bajo coste que no requiera equipo de cómputo especializado para evitar aumentar el costo de la tarea inspección.
- c. Determinar el hardware y las diferentes configuraciones a utilizar por el dispositivo para la adquisición de imágenes 2D de componentes aeronáuticos instalados en ala con acceso limitado.
- d. Crear una solución genérica que pueda fácilmente utilizarse en otros sectores que requieran la ejecución de pruebas de inspección visual.

## **1.6 HIPÓTESIS**

Empleando modelado 3D se puede crear una representación automática de bajo coste para realizar pruebas de inspección visual en componentes aeronáuticos, como la cámara de combustión, que presentan limitaciones de acceso cuando se encuentran montados en el ala de la aeronave.

## **2. MARCO TEÓRICO**

En este capítulo se presentan los conceptos básicos necesarios para el entendimiento del desarrollo de este proyecto. Se abarca el tema de los sistemas de inspección, describiendo las principales pruebas utilizadas en el sector aeronáutico y mostrando algunos ejemplos de sistemas equivalentes en otros sectores industriales.

De igual forma se describen conceptos básicos relacionados a la fotogrametría y reconstrucción 3D, donde también se resumen algunos de los principales programas existentes en la actualidad para llevar a cabo esta tarea.

### **2.1 PRUEBAS NO DESTRUCTIVAS (PND)**

Como parte del mantenimiento que se le realiza a una turbina se encuentran las pruebas no destructivas. Estas pruebas se caracterizan por no dañar el objeto a analizar. La evolución de la tecnología y su implementación cada vez más común en la industria han llevado a la simplificación de este tipo de actividades e inclusive en algunas ya no es requerida la interacción humana para su ejecución. Estas pruebas son características en procedimientos de inspección, dentro de los más comunes podemos encontrar pruebas ultrasónicas, pruebas de termografía, pruebas basadas en vibración, inspección visual, entre otras<sup>[1]</sup>.

#### **2.1.1 Pruebas ultrasónicas**

Consisten en el uso de ondas de sonido (generalmente de un orden de 1–50 MHz) para la detección de defectos en un componente. Este tipo de técnicas consisten en el uso de un transmisor y un receptor a través de los cuales se mide el tiempo de transmisión de onda, la velocidad de propagación de onda, la amplitud de ésta y la pérdida de energía debido a la atenuación de onda. Algunos de los métodos más comunes de este tipo de técnica son: transmisión, reflexión y retro dispersión<sup>[2]</sup>.

La ventaja de este tipo de técnica es la velocidad y resolución para poder detectar defectos, así como su habilidad para ser usada en campo. Sus desventajas, por otro lado, son la dificultad para preparar la prueba, la necesidad de contar con muestras

para asegurar la precisión de la prueba y un alto conocimiento para examinar correctamente el componente<sup>[2]</sup>.

### 2.1.2 Termografía infrarroja

Esta técnica está basada en la adquisición de una imagen visual a través de la inyección de radiación infrarroja en un objeto, logrando de esta forma obtener una representación visual para su posterior análisis. Algunos de los dispositivos más avanzados ya incluyen una serie de sensores para producir una imagen infrarroja de un escenario específico. Una vez obtenida la imagen infrarroja, se transforma a una imagen visual donde a cada nivel térmico localizado se le asigna un color determinado, como se puede observar en la Figura 3<sup>[3]</sup>. Entre sus principales ventajas están la capacidad de inspeccionar superficies grandes sin la necesidad de acoplar partes adicionales, como sensores, al objeto a inspeccionar. Por otro lado, algunas de las desventajas que presenta es que la instrumentación requerida para llevar a cabo la inspección suele ser muy cara y requiere de personas experimentadas para su uso. Finalmente, es capaz de detectar sin problema alguno aquellos defectos localizados en la superficie o en capas cercanas a ella. Sin embargo, si el deterioro se encuentra en un punto muy profundo, los resultados obtenidos por este tipo de inspección podrían no ser muy confiables<sup>[2]</sup>.

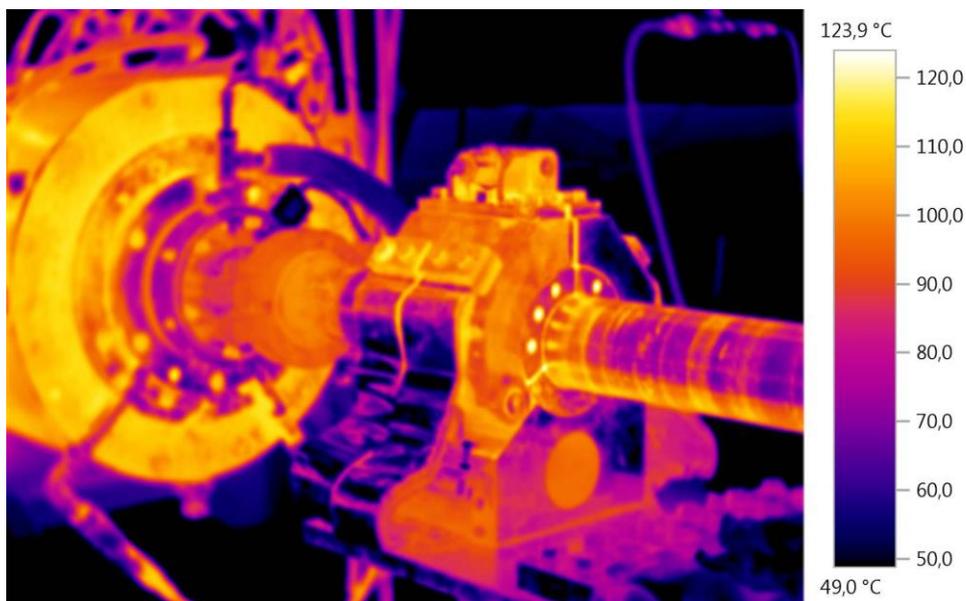


Figura 3. Termograma Infrarrojo realizado a la flecha de una turbina representando sus partes de acuerdo con la temperatura detectada en los mismos<sup>[48]</sup>.

### 2.1.3 Pruebas basadas en vibración

Las pruebas basadas en vibración resultan efectivas para determinar cómo responderá un componente en su uso cotidiano. Estas pruebas consisten en someter el componente ante diferentes efectos, por ejemplo, sacudirlo, golpearlo, entre otros, simulando la realidad una vez que se encuentre en operación. Este método se caracteriza por su alta sensibilidad y rapidez de inspección al usar múltiples sensores. Al igual que las pruebas ultrasónicas, se requiere mucha instrumentación para su correcta ejecución. La Figura 4 muestra al CHEOPS (CHaracterizing ExOPlanet Satellite) siendo sometido a pruebas estructurales a través del uso de una mesa de agitación o *Shaker*. Un *Shaker* se compone de una plataforma vibrante, un bastidor que es una estructura fija, unos tacos de goma y un motovibrador localizado debajo de la plataforma vibrante. El componente por inspeccionar se coloca en la plataforma vibrante (área negra); posteriormente, se acciona el motovibrador, el cual con la ayuda de los tacos de goma comenzará a posicionar la plataforma en diferentes ángulos a altas velocidades definidas por el inspector, haciendo vibrar al componente. Mientras el componente se encuentra vibrando, se captura la información de la fase y frecuencia de vibración del componente. En industrias como la aeronáutica, donde la flecha principal de una turbina es un eje rotatorio, se desea que dicho eje gire suavemente. A través de las pruebas de vibración es posible identificar problemas con estas partes rotativas para prevenir daños estructurales del componente<sup>[24]</sup>.



Figura 4. Posicionamiento del CHEOPS en una mesa de agitación para realizar pruebas de estructurales del satélite<sup>[41]</sup>.

## 2.1.4 Inspección visual

Es el método más común en ciertas pruebas por su bajo costo. Como su nombre lo dice, consiste en el análisis visual de una parte por medio de un operador u otro equipo para determinar la presencia de algún defecto. Además de su bajo costo, no requiere equipo especializado para su ejecución, o el equipo requerido no suele ser muy costoso. Un ejemplo de este tipo de inspección es a través del uso del boroscopio, el cual es usado para obtener una imagen visual de las partes internas de un sistema que presenta limitaciones para poder acceder a ellos<sup>[9]</sup>.

Usualmente estas actividades son llevadas a cabo por técnicos, entrenados en principio en el aspecto teórico de las técnicas de inspección, así como la clasificación en la cual pueden caer cada uno de los defectos encontrados. La Figura 5 muestra un equipo de inspección boroscópica, donde los inspectores usan técnicas precisas para lograr el correcto desplazamiento del boroscopio para tener una captura adecuada del medio inspeccionado. Sin embargo, es común encontrar diferencias entre lo aprendido y lo requerido ya en la estación de trabajo. Es en este punto donde técnicos con mayor experiencia ayudan a técnicos novatos a identificar y clasificar debidamente un defecto<sup>[8]</sup>.

La falta de claridad debido a capturas inadecuadas del escenario inspeccionado ha llevado a buscar nuevas formas para realizar este tipo de actividades, con el objetivo de eficientizar tiempos, reducir riesgos derivados por falta de experiencia ante una situación específica y siempre salvaguardar la seguridad de las personas que utilizan los aviones como medio de transporte.



Figura 5. Equipo para inspección boroscópica con pantalla para visualización<sup>[8]</sup>.

Los sistemas de inspección anteriormente mencionados fueron analizados para determinar su viabilidad como alternativa para este proyecto. A través de este análisis se logró determinar que la inspección visual es la alternativa más viable para la ejecución de este proyecto, por ser un método simple y el cual puede ser adaptado fácilmente a diferentes aplicaciones. La Tabla 1 resume aquellos sistemas que fueron considerados en un principio y la justificación correspondiente de por qué no fueron seleccionados.

*Tabla 1. Análisis comparativo de sistemas de inspección en la industria aeronáutica.*

<b>Sistema de Inspección</b>	<b>Considerado</b>	<b>Seleccionado</b>	<b>Justificación</b>
Ondas Ultrasónicas	Si	No	Requiere instrumentación especializada
Termografía Infraroja	No	No	Costo, conveniente para superficies, mas no para partes internas
Vibración	No	No	Enfocada a partes rotativas
Visual	Si	Si	Método simple y de fácil adaptación

### **2.1.5 Robots para automatización de pruebas no destructivas**

Algo que caracteriza los diferentes métodos mostrados anteriormente para la realización de pruebas no destructivas es la necesidad, en su mayor parte, de un operador encargado ya sea de realizar la prueba o del análisis de los resultados. El uso de la tecnología en estos procesos poco a poco ha logrado llegar a un punto donde estas pruebas pueden ser realizadas de manera automática o semiautomática, obteniendo resultados cada vez más precisos. Todo esto ha llevado al estudio y desarrollo de nuevas técnicas para poder ejecutar este tipo de pruebas en un menor tiempo.

Hoy en día es común escuchar hablar de robots que realizan ciertas actividades humanas. Este tipo de aplicaciones ha comenzado a adentrarse en los diferentes sectores de la industria, teniendo hoy en día robots encargados de algunas de estas actividades. En algunas líneas de producción ya podemos observar a robots inspeccionando la calidad del producto y determinando si cumple o no con los estándares definidos.

## 2.2 SISTEMAS DE INSPECCIÓN

Cuando se habla de sistemas de inspección, uno de los principales conceptos con los que se relaciona es con calidad. Sin embargo, existen otros objetivos de estas pruebas con otros enfoques como son la determinación y prevención de fallas de algunos componentes. La principal razón de estas pruebas de inspección es el poder tener en la manera de lo posible el control del tiempo enfocado al mantenimiento del componente. En términos industriales, el control es un sinónimo de operación. Si tomamos como ejemplo una máquina de una línea de producción, no es lo mismo tener calendarizado el mantenimiento requerido, a tener que parar la máquina repentinamente por una falla encontrada.

### 2.2.1 Inspecciones visuales en ingeniería civil

Diferentes aplicaciones de ingeniería han requerido una serie de técnicas basadas en teoría de procesamiento de imágenes, a través de las cuales se logra detectar fisuras en superficies. Un ejemplo es el que muestra la Figura 6, un vehículo autónomo sin piloto (UAV). Este tipo de vehículo ha sido utilizado en ingeniería civil para poder monitorear y dar mantenimiento a edificios, puentes y otros tipos de construcciones. Su tarea específica es realizar monitoreo a aquellas partes estructurales a las que no es sencillo llegar para identificar grietas u otras deficiencias en las superficies del objeto<sup>[10]</sup>.



Figura 6. Vehículo autónomo sin piloto con cámara para inspección de estructuras<sup>[10]</sup>.

Otro ejemplo es el sistema de transporte colectivo metro. De igual forma que los UAVs, el objetivo es el de encontrar deficiencias en la estructura por la cual se desplaza el metro para proceder a las reparaciones necesarias a tiempo. En este tipo de inspecciones, debido a que la estructura ya es recorrida por un vehículo (el metro), se le realizan

adaptaciones al mismo para la instalación de la instrumentación necesaria para la correcta adquisición de imágenes<sup>[13]</sup>.

Un ejemplo más en ingeniería civil es el descrito por Stent<sup>[45]</sup> para la realización de inspección de túneles a bajo costo. Este proyecto comenzó contando con una sola cámara para la captura de imágenes, pero con el paso del tiempo fue evolucionando, logrando ser capaz de resolver problemas previamente encontrados. El proyecto comenzó con tan solo una cámara teniendo limitantes de ángulo de captura (pues solo se lograba capturar en un rango de 180°) y la necesidad de operación manual de la misma. Posteriormente, se decidió añadir una segunda cámara y se incluyeron ejes rotativos para facilitar la captura del escenario en un rango de 360°. Sin embargo, se encontraron con problemas de estabilidad y alimentación de energía; además tras la adición de una cámara, el costo se elevaba y era necesario calibrar mejor los lentes de las mismas. El tercer intento y prototipo final consistió en la incorporación de un motor rotativo capaz de ser controlado electrónicamente y la eliminación de una cámara. De esta forma, la flexibilidad del dispositivo era más alta, se requería menos calibración del sistema para la adquisición de imágenes al tener una sola cámara y también se observó mayor calidad en las imágenes debida a una mejor iluminación en el túnel. La Figura 7 muestra el prototipo final descrito por Stent que fue utilizado para la inspección de un túnel de Londres.



Figura 7. Sistema de bajo costo para inspección visual eficiente de túneles<sup>[45]</sup>.

Los casos descritos anteriormente son ejemplos de inspección de estructuras ejecutados de forma manual. La determinación final basada en los resultados encontrados a través de las imágenes dependía de la percepción del inspector, donde en ocasiones, la dificultad por llegar a aquellas zonas afectadas, la carencia de luz adecuada entre otros factores, representaban limitaciones y deficiencias en el proceso de inspección al ser llevadas a cabo de esta manera.

En respuesta a esas limitaciones se buscaron soluciones para optimizar el desarrollo de estas actividades. Estas soluciones se basaron en el procesamiento de imágenes para la detección de los defectos en las superficies correspondientes utilizando técnicas de adquisición y procesamiento de imágenes.

### **2.2.2 Inspecciones de turbinas aeronáuticas**

Las turbinas que le permiten volar a un avión requieren llevar a cabo una serie de procedimientos de mantenimiento que aseguren su correcto desempeño. El mantenimiento de la turbina puede estar agendado tras haber transcurrido un número determinado de horas en vuelo o puede realizarse tras la detección de comportamientos inesperados en sus sensores y actuadores durante su desempeño. Algunos de estos mantenimientos pueden requerir el desmontaje de la turbina del avión. Esto no significa que el avión quede inhabilitado, puesto que estos procedimientos suelen llevarse a cabo con el replazo temporal de la turbina en lo que ésta se encuentra en el mantenimiento correspondiente. Este procedimiento puede resultar muy costoso cuando se llega a la conclusión de que la turbina no requería ningún tipo de mantenimiento. Por esta razón, parte de algunos procedimientos consisten en realizar una valoración previa para determinar la necesidad de un mantenimiento. Estos procedimientos tratan, siempre que sea posible, de realizarse con la turbina montada en el avión (*On-Wing*) con el objetivo principal de evitar el desmontaje<sup>[23]</sup>.

Por la simplicidad que requiere, la inspección visual es un método muy utilizado pues además de ser una opción económica, también permite poder analizar partes a las cuales es difícil acceder. En el sector aeronáutico se realizan inspecciones visuales empleando un boroscopio que gracias a su tamaño puede ser utilizado para observar el estado de las partes dentro de la turbina. En el caso particular de esta investigación

se crea una representación 3D del estado actual de la cámara de combustión, la cual puede ser interpretada por un técnico especialista para determinar la necesidad de proceder con algún tipo de mantenimiento.

### **2.2.3 Mantenimiento de motores en aeronaves**

En el sector de aviación comercial, las agencias reguladoras como la FAA o EASA definen las regulaciones a cumplir para aquellos componentes utilizados en una aeronave, para que éstos sean seguros y confiables<sup>[38]</sup>. En el caso específico de las turbinas de aviación, uno de los métricos utilizados para determinar la confiabilidad es la cantidad de veces que la turbina ha requerido mantenimiento (conocido como Shop Visits o SV) en un tiempo determinado<sup>[23]</sup>. Si bien es cierto que, al igual que los automóviles, es necesario darle un mantenimiento a la turbina cada determinado número de horas de vuelo también es cierto que no todas estas visitas son siempre necesarias.

Es muy importante conocer los métricos operacionales de las aerolíneas para identificar sus necesidades actuales. Uno de estos métricos es la cantidad de horas que una aeronave se mantiene en operación continua. La Figura 8 muestra los beneficios económicos de las aerolíneas de 2005 a 2018, dichos beneficios se han incrementado significativamente desde el año 2015 en comparación con los 10 años previos, permaneciendo en un nivel constante. Es importante saber que esto ha sido posible a los tiempos de operación de las aeronaves, los cuales son directamente afectados en caso de remociones no necesarias de una turbina. Dichas remociones innecesarias son llevadas a cabo debido a la falta de un método de confirmación de la falla con la turbina instalada, implicando altos costos a la aerolínea<sup>[4]</sup>.

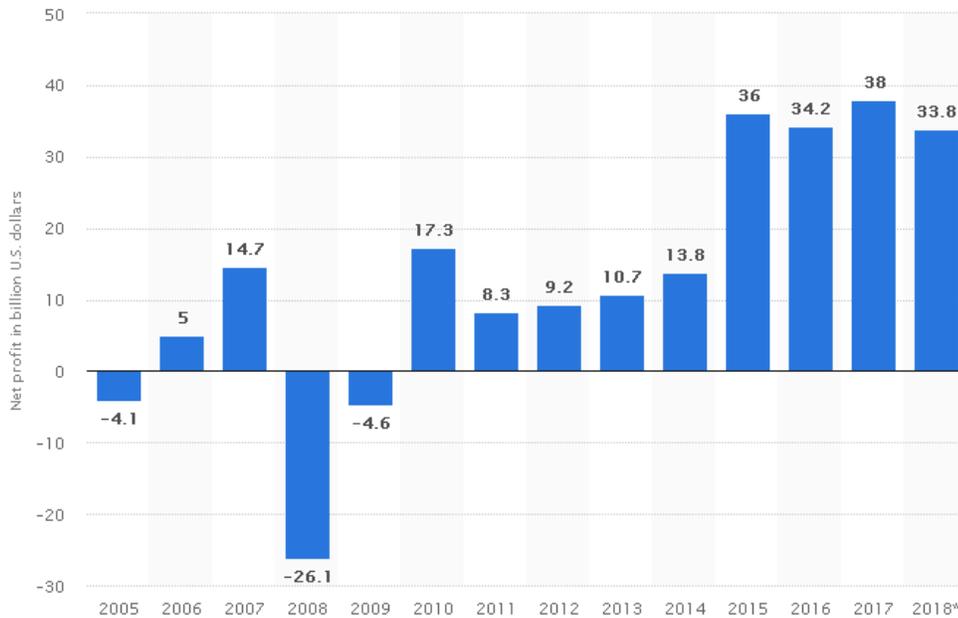


Figura 8. Ganancia neta de las aerolíneas de 2015 al 2018<sup>[4]</sup>.

El mantenimiento que se le puede realizar a una turbina de avión está clasificado en dos tipos principales: mantenimiento correctivo y preventivo. El mantenimiento preventivo es aquel que se tiene planeado realizar a una turbina al cumplirse una condición específica: horas de operación, distancia recorrida, entre otros. Este mantenimiento abarca actividades de lubricación, monitoreo de datos referentes a componentes específicos, así como actividades de inspección para la prevención de posibles fallas<sup>[5]</sup>.

Por otro lado, el mantenimiento correctivo es aquel que se ejecuta después de haberse detectado una falla en algún componente. Es necesario realizar el análisis de estas fallas para validar que el motor se encuentra en condiciones aceptables de operación y de esta manera asegurar el correcto funcionamiento del avión. Dependiendo del tipo de falla y su nivel crítico pueden existir acciones de mantenimiento a realizar, implicando en ocasiones la remoción del motor y su envío al taller. Este tipo de acciones inesperadas suelen tener un alto impacto en las aerolíneas, ya que, dependiendo de muchos factores, podrían llegar a significar la reducción de su flota por un tiempo indefinido en caso de no contar con un motor sustituto<sup>[5]</sup>.

## 2.2.4 Sistema de pronóstico y salud del motor

El principal objetivo de este tipo de sistemas es estimar el estado de salud actual de los componentes y determinar cuándo se espera una falla de éstos, con el objetivo de tomar acciones preventivas. La Figura 9 muestra las ventajas principales que le proveen a las aerolíneas estos sistemas, como: mejoras en la planeación de mantenimiento, reducción de mantenimientos no agendados, mejora en la identificación de causas raíz de un problema, entre otras<sup>[6]</sup>.



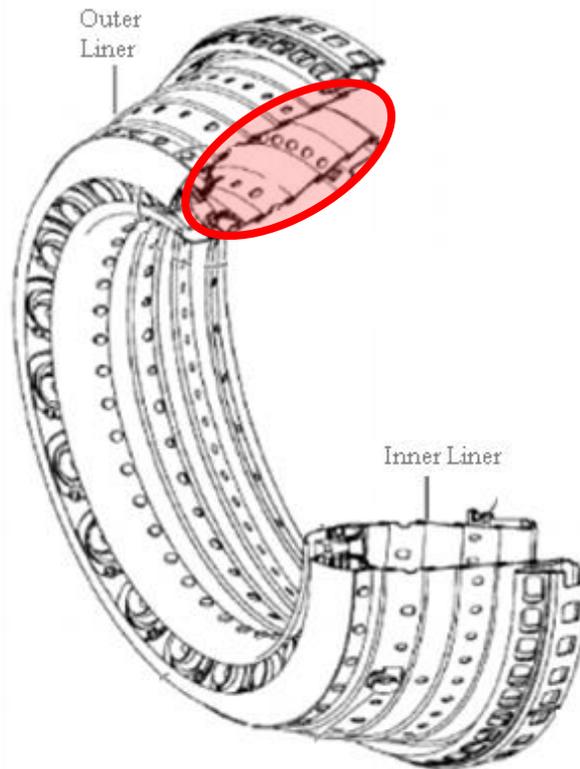
Figura 9. Principales beneficios de los sistemas de pronósticos <sup>[6]</sup>.

Para lograr su objetivo, el sistema de pronóstico y monitoreo de salud requiere adquirir, analizar y procesar información de los diferentes componentes que integran la turbina. Toda esta información proviene de los diferentes sensores y actuadores que la integran a través de los cuales es posible determinar las condiciones de operación actual de la máquina. En muchas ocasiones se comparan estas condiciones con aquellas operadas durante una etapa específica de vuelo. Para determinar la salud de algunos componentes, en ocasiones es tan sencillo como tomar la información actual y compararla contra un límite fijo previamente definido por un modelo. La Figura 10 es una representación de la detección de una falla. Se puede observar en la figura que cruzar este límite significa que se ha salido del rango de operación esperado y por lo tanto se indica una falla para que el operador la pueda atender<sup>[6]</sup>.



La cámara de combustión es uno de los componentes más críticos de una turbina. Está formada por superaleaciones basadas en cobalto y níquel, mezcladas con aluminio y titanio para fortalecer la estructura, y otros componentes, como puede ser el cromo, para hacerla resistente a la corrosión<sup>[46]</sup>. El principal objetivo de este componente es el de usar eficientemente el combustible para generar la energía necesaria para el desplazamiento de la aeronave. A la cámara llega aire comprimido en etapas anteriores del sistema de la turbina, el cual es mezclado con combustible para de esta manera generar la energía necesaria por la turbina<sup>[7]</sup>. Debido a las altas temperaturas a las que se encuentra este componente, es difícil contar con instrumentación que nos ayude a saber con exactitud si la cantidad de aire y de combustible que ingresaron a la cámara eran los calculados para la etapa de compresión y combustible. Aunque es posible identificar algún problema, ya que una mala combustión implicaría una generación de energía menor a la deseada o un rendimiento de combustible menor. Sin embargo, este tipo de problemas contienen demasiadas variables que le dificultan al operador identificar el componente que presenta la causa raíz del problema.

Al ser la cámara de combustión un componente tan importante, es necesario asegurar su correcto funcionamiento todo el tiempo que se encuentre en operación. La Figura 12 muestra la parte de la cámara donde se genera la combustión. El espacio entre la capa superior, descrita en la figura como *Outer Liner*, y la capa inferior, descrita como *Inner Liner*, es el espacio a analizar. Este espacio, de forma rectangular, es recorrido por una cámara controlada por un operador experto para su estudio, con el objetivo de determinar la salud del componente mientras la inspección se lleva a cabo.



*Figura 12. Cámara de combustión de un motor CFM56. Generalmente la inspección boroscópica se realiza recorriendo el Inner Liner<sup>[33]</sup>.*

Todo esto se requiere realizar de una manera eficaz con la turbina instalada en el ala del avión, evitando de esta manera los tiempos operativos correspondientes a la remoción y desplazamiento de la turbina a un taller de mantenimiento. La Figura 13 muestra la imagen obtenida durante una inspección boroscópica de la cámara de combustión. Como se puede observar en la Figura 13, el interior de la cámara de combustión es de un color rojizo, aunque este puede variar de acuerdo a las aleaciones utilizadas por cada proveedor. La cámara presenta una serie de orificios por medio de los cuales se realiza la inyección de combustible y de aire para la combustión. A lo largo de su vida, el componente ha presentado manchas negras relacionadas con la combustión del combustible y debido a sus altas temperaturas es posible que comience a presentar grietas, como lo muestra la Figura 13. Durante la ejecución de actividades de mantenimiento, principalmente, se limpia el componente para remover obstrucciones encontradas en los inyectores y en caso de ser necesario se pueden realizar reparaciones a las fisuras encontradas, lo que implica soldar pequeñas partes de la cámara con la misma aleación de la que se compone.



Figura 13. Grietas observadas dentro de una cámara de combustión<sup>[49]</sup>.

### 2.2.5 Tendencias en inspección de componentes aeronáutico

Las diferentes necesidades como la ejecución de pruebas de inspección en ala, así como las limitaciones sobre áreas de acceso para la correcta visualización de partes internas de la turbina, como la cámara de combustión, han llevado a la definición de alternativas a métodos convencionales de inspección para su ejecución. En el sector aeronáutico se ha iniciado una etapa revolucionaria con la creación de nuevos dispositivos para dichas actividades. El Dr. James Kell, especialista de tecnología de la compañía Rolls Royce, dio a conocer que la empresa se encuentra desarrollando dispositivos de 10 mm de diámetro, véase la Figura 14, que permitan ingresar a componentes dentro de la turbina para su análisis y reparación<sup>[20]</sup>.



Figura 14. El robot 'cucaracha' de 10 mm de Rolls Royce tiene por objetivo simplificar actividades de inspección y reparación en componentes aeronáuticos<sup>[20]</sup>.

En términos de inspecciones requeridas para componentes externos, también se han llegado a diseñar prototipos para estas tareas. Un ejemplo de esto es el mostrado en la Figura 15 que tiene el principal objetivo de determinar curvaturas no esperadas en el fuselaje de una aeronave, un análisis requerido periódicamente por agentes reguladores. Para lograr lo anterior, se utiliza un escáner, sensores de carga y copas de succión en los soportes del robot. Mientras el robot se desplaza a lo largo de las superficies, las copas de succión en cada soporte del robot determinan la estabilidad de cada soporte, en base a lo cual se logra determinar alguna curvatura o fisura en la estructura de la aeronave [47].



*Figura 15. Robot de desplazamiento por principio de succión utilizado para análisis y determinación de curvaturas en estructuras aeronáuticas [47].*

En la actualidad, el uso de nuevas tecnologías y la creación de nuevas técnicas de inspección se han convertido en objetos de estudio en el sector aeronáutico, buscando la automatización de estos procesos, sobre todo de aquellos que requieren una alta logística para su ejecución. Sin lugar a duda, la propuesta presentada por Rolls Royce revolucionará ciertas actividades de inspección. Sin embargo, existen productos con limitaciones de acceso donde se necesitan componentes más pequeños a ser introducidos, por ejemplo, dentro de una turbina. Esto es posible conseguirlo a través de los robots suaves por la capacidad de movilidad que presentan[52].

Mientras que los materiales de este tipo de robot es lo que los caracteriza, es necesario crear un sistema capaz de controlar los diferentes componentes de los mismos para

poder llevar a cabo la ejecución de estas tareas de inspección. Este sistema debe presentar una alta modularidad, que permita la adición de nuevas funcionalidades constantemente para ir cubriendo nuevas necesidades del mismo. Otro punto importante a considerar es que la ejecución de las tareas de inspección requiere de un operador altamente familiarizado con el objeto de inspección, pues durante estas tareas se determina el estado de dicho objeto y la necesidad de proceder con algún tipo de mantenimiento. Sin embargo, esto se ha convertido en una restricción pues se depende de la disponibilidad de dicho operador, creando tiempos de espera en suelén ser altamente costosos. Separando la tarea de inspección en dos partes es posible eliminar estos tiempos de espera. En primer lugar, se tiene la tarea enfocada a la captura de información del objeto de estudio, la cual puede ser realizada por un robot de manera automática. En segundo lugar, se encuentra el análisis de dicha información para la determinación de su estado actual, la cual en una etapa inicial requeriría de un operador. Sin embargo, tras contar con suficiente información podría ser posible automatizarla.

## **2.3 FOTOGRAMETRÍA Y RECONSTRUCCIÓN EN 3D**

La fotogrametría es una técnica cuyo objeto es estudiar y definir con precisión la forma, dimensiones y posición en el espacio de un objeto cualquiera utilizando esencialmente medidas hechas sobre una o varias fotografías de ese objeto<sup>[25]</sup>. También es considerada como la técnica que permite realizar medidas 3D y obtener volúmenes a partir de fotografías<sup>[26]</sup>.

Existen dos principales divisiones de la fotogrametría: terrestre y aérea, cada una está caracterizada por la posición de la cual se adquirió la fotografía. De esta manera, la fotogrametría terrestre es aquella donde las imágenes fueron adquiridas de manera horizontal o paralelas al plano, mientras que la fotogrametría aérea es aquella donde las imágenes se adquieren de manera vertical o perpendicular al plano<sup>[25]</sup>.

### **2.3.1 Triangulación pasiva**

Este es uno de los principios básicos para la reconstrucción 3D, así como uno de los más utilizados. Dadas dos proyecciones sobre el mismo punto en dos imágenes, su posición en el plano tridimensional se encuentra en aquel punto donde ambas proyecciones

interceptan. La repetición de estas proyecciones para una serie de puntos distintos llevará a la conformación de la forma en 3D del objeto. Visto de otra manera, es como tener dos cámaras monitoreando el mismo objeto, como se muestra en la Figura 16. Los fundamentos de este principio radican en encontrar un punto específico de una imagen en la segunda<sup>[17]</sup>.

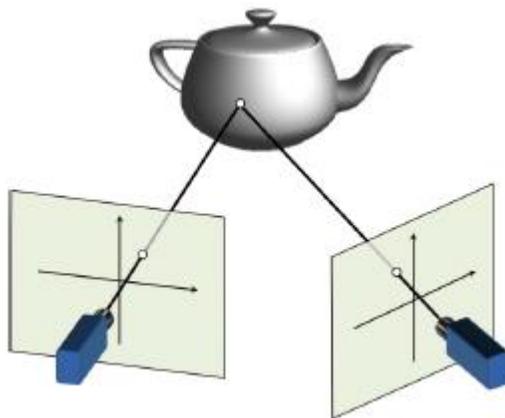


Figura 16. El principio de triangulación pasiva se basa en la comparación de puntos de un objeto visto desde dos perspectivas diferentes<sup>[17]</sup>.

La mayoría de las aplicaciones de reconstrucción 3D a partir de imágenes consideran la adquisición de las imágenes posicionando la(s) cámara(s) alrededor del objeto de estudio. Este proceso es prácticamente realizar un escaneo del objeto, mientras existan mayor cantidad de puntos de enfoque del objeto, será posible obtener mayor detalle en el modelo 3D<sup>[17]</sup>.

### 2.3.2 Agrupación de imágenes

En procesamiento de imágenes, la función de agrupar imágenes para formar imágenes de mayor tamaño es conocida también como *stitching*. Las bases para este método son muy similares a la triangulación pasiva; se busca principalmente detectar aquellos puntos que son compartidos entre dos imágenes para posteriormente sobreponer aquellas áreas similares y de esta manera coser las imágenes para obtener una sola.

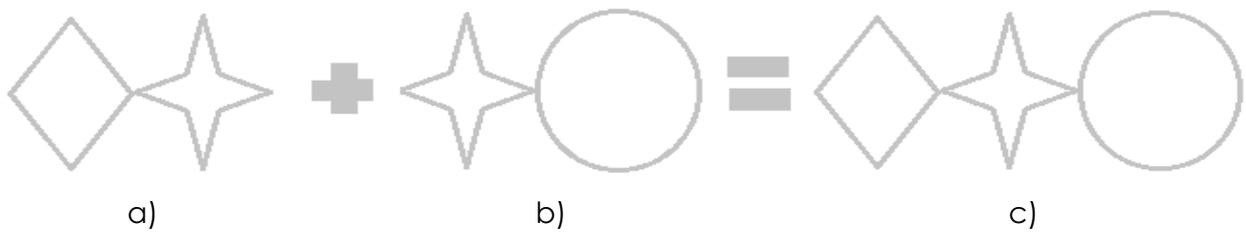


Figura 17. El stitching de imágenes (c) se logra tras la sobre posición de aquellas regiones comunes entre diferentes imágenes (estrella en a y b).

Un software que está teniendo mucho auge en el procesamiento de imágenes, sobre todo para aplicaciones médicas, es *ImageJ*<sup>[39]</sup>. Una versión paralela basada en este software, pero la cual ya posee múltiples *plug-ins* se conoce como *Fiji*<sup>[53]</sup>. Uno de los *plug-ins* ofrecidos por esta aplicación es el de *Stitching*, desarrollada por Stephan Preibish<sup>[37]</sup>. Esta utilidad ofrece diversas opciones para integrar múltiples imágenes en una sola, como las mostradas en la Figura 18. Una de las ventajas de la utilidad de *Stitching* es su versatilidad; es posible indicar la forma en cómo deseas realizar la agrupación (de manera vertical, horizontal, por columnas, por nombre de imágenes, etc.).

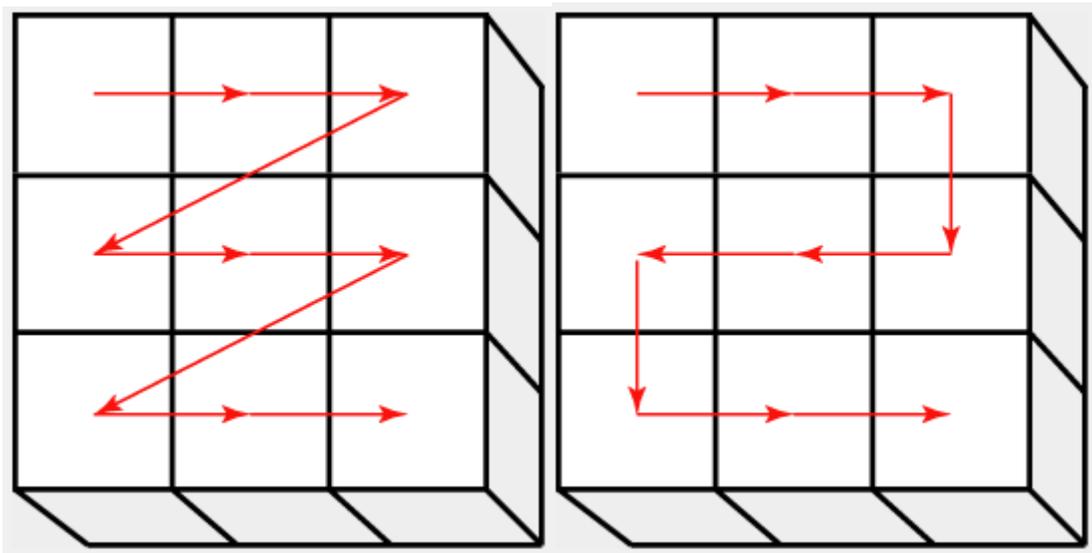


Figura 18. Configuraciones ofrecidas en Fiji para realizar el *Stitching* de múltiples imágenes<sup>[37]</sup>.

### 2.3.3 Reconstrucción 3D con imágenes 2D

La reconstrucción de modelos 3D a partir de imágenes 2D es posible a través de las cinco funciones descritas en la Figura 19.

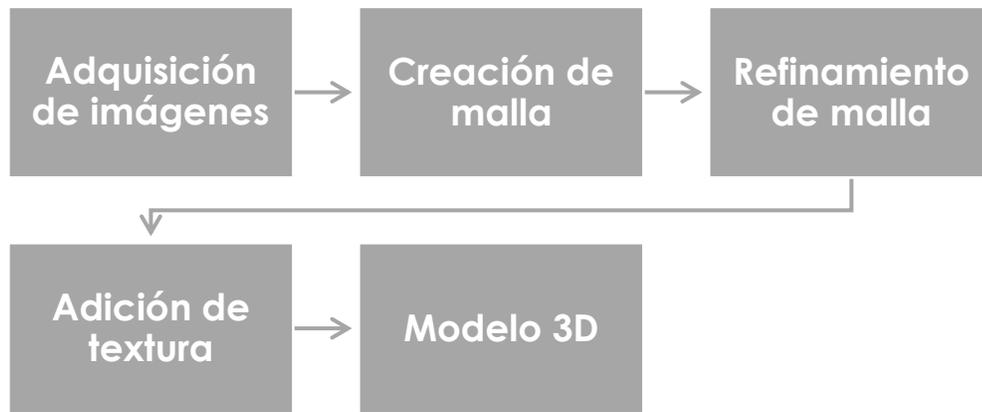


Figura 19. Pasos por seguir para una reconstrucción en 3D con imágenes 2D<sup>[29]</sup>.

### a. Adquisición de imágenes

Es la tarea de tomar las fotografías necesarias del objeto de estudio. Algunas consideraciones para tomar en cuenta es la cantidad de imágenes requeridas. Es importante señalar que, entre mayor cantidad de imágenes, así como mayor resolución, la reconstrucción de 3D mostrará mejor detalle del objeto. De acuerdo con autores como Moons se recomienda tomar por lo menos 9 imágenes<sup>[17]</sup>. Sin embargo, en promedio podemos observar alrededor de 50 imágenes en investigaciones como la de como Falkingham<sup>[29]</sup>. Es importante mencionar que investigaciones como la de Falkingham se basan en la reconstrucción de un objeto en específico y no de una escena como la que se requiere para reconstruir una inspección boroscópica. La Figura 20 muestra la entrada de una cripta bizantina que fue estudiada y analizada. Se realizaron múltiples fotografías de la misma para posteriormente utilizarlas en la etapa de reconstrucción.



Figura 20. Entrada a cripta bizantina<sup>[27]</sup>.

## b. Creación de malla

Se le denomina malla a la representación de un objeto por medio de polígonos. Consiste principalmente de tres partes: vértices, bordes y caras<sup>[30]</sup>. Un vértice representa la posición de un punto específico del objeto de estudio; un borde es la unión que existe entre dos vértices; y una cara es un conjunto cerrado de bordes que forman un polígono. Con las imágenes previamente capturadas se identifican las diferentes caras del objeto estudiado. La Figura 21 muestra la creación de la malla, la cual se realiza al unir todas las caras detectadas dentro de la imagen.

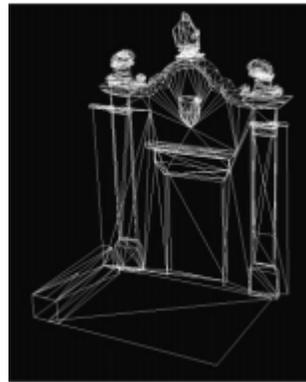


Figura 21. Malla de la entrada a cripta bizantina<sup>[27]</sup>.

## c. Refinamiento de malla

El refinamiento de la imagen consiste en subdividir los polígonos que conforman la malla actual, en polígonos más pequeños con el objeto principal de obtener mejor detalle en las áreas deseadas para su representación 3D (ver Figura 22). En ocasiones, el refinamiento de la malla no es necesario, e inclusive cuando se requiere, suele limitarse a solo aquellas partes del objeto que se requieren estudiar. Todo esto debido al alto costo computacional que implica la subdivisión de la malla<sup>[31]</sup>.

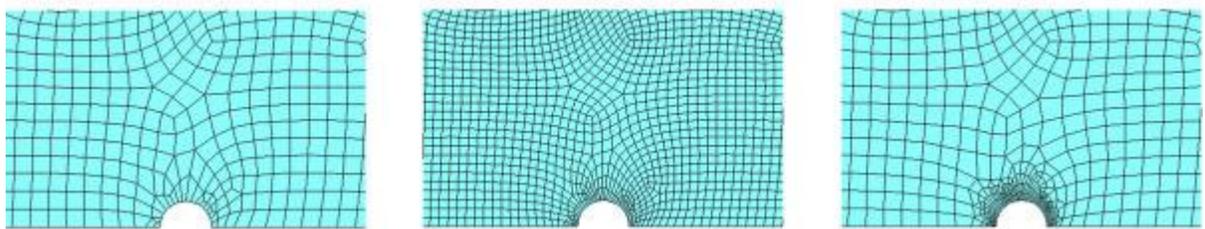


Figura 22. La figura de la izquierda es la malla creada. La figura del centro es el refinamiento total de la malla. La figura de la derecha muestra un refinamiento local en la parte inferior central<sup>[31]</sup>.

#### **d. Adición de textura**

Para concluir el proceso de reconstrucción en 3D, una vez que ha sido creada y refinada la malla se le agrega una textura para mejorar su representación. En la Figura 23 es posible observar como la textura otorga un nivel de detalle a la imagen para asemejarla lo más posible a la realidad. La textura es insertada en caras específicas de la malla dependiendo del modelo 3D deseado.



Figura 23. Ejemplo de texturización de una malla<sup>[27]</sup>.

Una vez concluidos los pasos anteriores se obtiene el Modelo 3D.

#### **2.3.4 Escaneo láser**

Existen aplicaciones donde se desea estudiar un objeto desde su interior, más que desde el exterior. Para estos casos una de las tecnologías más utilizadas es el escaneo láser. Este escaneo está basado en la técnica *LiDAR (light detection and ranging)*. Esta técnica requiere un transmisor/receptor de rayo láser, un dispositivo de escaneo y un dispositivo temporizador. El escaneo láser de un objeto consiste en emitir un rayo láser o una serie de pulsos para que posteriormente el dispositivo de escaneo reciba y registre la señal reflejada. El temporizador toma la cantidad de tiempo que le tomó a cada pulso realizar el recorrido y con esta información es posible calcular la distancia. Conociendo el ángulo y dirección del rayo emitido es posible determinar la posición relativa de la superficie del objeto analizado. De esta forma es posible obtener una nube de puntos manipulando el ángulo del rayo. Una vez adquirida esta información, es posible recrear el entorno donde se encuentra el láser<sup>[28]</sup>. La Figura 24 muestra este principio.

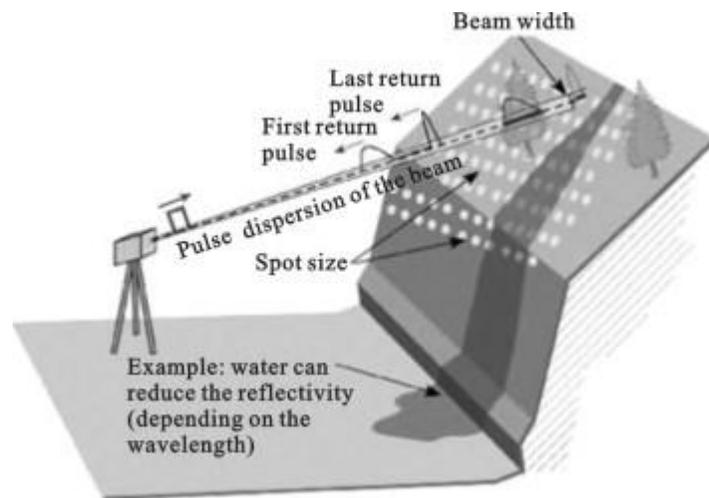


Figura 24. Principio de adquisición de información con un rayo láser<sup>[28]</sup>.

La mezcla de ambas técnicas (fotogrametría y escaneo láser) son una forma muy poderosa de recrear escenarios en ambientes 3D. La Figura 25 muestra un ejemplo que podemos encontrar en el estudio realizado por Beraldin<sup>[27]</sup>, a través del cual utilizan el escaneo láser para obtener una representación tridimensional de una cripta bizantina, la cual puede verse como una simple maqueta de ésta mientras que a través de la fotogrametría adquieren las texturas para darle una representación más realista al modelo.



Figura 25. A la izquierda el modelo 3D obtenido por escaneo láser. A la derecha el modelo 3D con la textura obtenida por fotogrametría<sup>[27]</sup>.

## 2.4 SOFTWARE

En la actualidad existen diferentes programas para construir modelos en 3D. A lo largo de esta investigación se encontraron tres principales clasificaciones de este tipo de software.

### 2.4.1 Reconstrucción en 3D

Los programas para la reconstrucción en 3D a partir de imágenes 2D deja al desarrollador la tarea de una buena adquisición de imágenes. En el portal del Dr. Peter Falkingham<sup>[29]</sup> podemos encontrar un análisis sobre estos programas tomando en consideración dos principales factores: tiempo de procesamiento y calidad del resultado. Estos programas en ocasiones suelen demandar un alto uso de memoria tanto de CPU como de GPU. De igual forma, se encontró que el tiempo de procesamiento está relacionado con la calidad del modelo obtenido. La Tabla 2 muestra la comparativa en términos de tiempo de procesamiento y calidad del modelo resultante de diferentes programas utilizados para reconstrucción en 3D.

*Tabla 2. Comparativa en tiempo de procesamiento y calidad del modelo 3D de diferentes programas para reconstrucción en 3D.*

	Tiempo de procesamiento	Calidad del Resultado
VisualSFM & OpenMVS	~20 min	Mala
AliceVision Meshroom	~65 min	Excelente
Colmap	~122 min	Excelente
3DF Zephyr Free	~33 min	Regular
OpenMVG + OpenMVS	~128 min	Buena

La relación entre tiempo de procesamiento y calidad del modelo es inversamente proporcional. Para la ejecución de una prueba visual es necesario otorgar un nivel de detalle aceptable en el menor tiempo posible. Por esta razón, el uso de estos programas no resulta viable ya que estos tiempos están basados en el modelo de una figura de cerámica. Sin embargo, todos estos programas coinciden en la creación de un modelo para posteriormente texturizarlo, en donde la primera tarea es la que les consume un mayor tiempo.

### 2.4.2 Modelado 3D

Los programas de modelado 3D son aquellos utilizados para la creación de objetos y personajes para ser usados en motores de contenidos 3D. Uno de los programas más utilizados en la actualidad es Blender<sup>[32]</sup>. Entre algunas de sus ventajas se encuentran que es un software open source, basado en Python, y el cual cuenta con mucha documentación debido al gran número de usuarios que lo utilizan. Igualmente, presenta

la posibilidad de automatizar la generación de modelos a través del uso de scripts en lenguaje Python.

### **2.4.3 Visualización de modelos y motores de contenidos**

De igual forma como existen los programas para la creación de modelos 3D, también existen diferentes programas para su visualización. En línea podemos encontrar Sketchfab (<https://sketchfab.com/>), una plataforma que permite a los diferentes desarrolladores compartir y vender los modelos que han creado. De igual forma, programas de modelado 3D como Blender contienen una configuración que permite enlazarse directamente a esta plataforma y publicar automáticamente los modelos creados por el software de modelado. La limitante de esta plataforma es que solo permite visualizar un modelo a la vez, es decir, no existe una forma para visualizar diferentes modelos contenidos en un escenario. Para estos casos, lo más apropiado es el uso de motores de contenidos 3D o de videojuegos. Un claro ejemplo de esto es Unity, un motor de videojuegos multiplataforma. Este trabajo está basado en el uso de Blender pues consiste en la creación de un solo modelo y no en un conjunto de modelos diferentes que requirieran un motor de contenido 3D.

## **2.5 INSPECCIÓN AUTOMÁTICA EN COMPONENTES AERONÁUTICOS**

En la actualidad las inspecciones de componentes aeronáuticos las realizan operadores con los conocimientos necesarios para determinar el estado actual del componente. Para lograr esto el operador debe saber cómo manejar el instrumento de medición utilizado para obtener la información que le permita determinar la necesidad de ejecutar alguna acción de mantenimiento. En ocasiones la aeronave se localiza en regiones donde no existen operadores capacitados. Esto genera la necesidad de trasladar a un operador o iniciar la remoción de la turbina para su traslado, agregando tiempos no operables de la turbina y de la aeronave, los cuales a veces son necesarios pues a través de la inspección visual se determina que no existe ningún problema con el componente. Es posible reducir estos costos separando la tarea de inspección en dos partes:

1. *Visualización del componente*, que consiste en la adquisición de información del estado interno de un componente obtenida generalmente por una cámara para su representación y visualización.
2. *Determinación del estado de un componente*, que consiste en interpretar la información visual obtenida para determinar la necesidad de ejecutar alguna acción de mantenimiento.

La primera parte de la tarea de inspección es posible automatizarla con tecnología actual. En la sección 2.2.5 se describieron algunas propuestas para realizar este tipo de inspecciones. Sin embargo, estas soluciones están enfocadas a trabajar con la turbina desmontada del ala. Este trabajo se enfoca en la realización de pruebas de inspección de componentes aeronáuticos sin necesidad de retirar del ala el componente que se desea inspeccionar.

### **3. PROCEDIMIENTO DE INVESTIGACIÓN**

En este capítulo se describirá el método propuesto para la reconstrucción de un escenario inspeccionado. Este método se basa en la adquisición de imágenes del escenario para posteriormente agruparlas en un modelo 3D que pueda ser utilizado por un inspector de manera remota para dictaminar la operabilidad del componente, y con base en esto determinar las actividades de mantenimiento correspondientes.

El método propuesto presenta una alternativa semiautomática para la ejecución de pruebas de visualización tomando en cuenta las siguientes consideraciones:

1. Proveer una solución de bajo costo que pueda sustituir la forma actual de realizar la captura de información de las inspecciones boroscópicas de componentes aeronáuticos con la turbina en ala, evitando de esta manera los costos derivados de su desmontaje y traslado a talleres de servicio.
2. Representar el escenario capturado en un sistema 3D que pueda ser visualizado por el operador para su valoración.
3. Describir el hardware y las configuraciones con las que puede contar un robot suave para la adquisición de imágenes.
4. Crear una solución genérica capaz de utilizarse en otros sectores que requieran pruebas de visualización.

#### **3.1 REPRESENTACIÓN DE ESCENARIO DE BAJO COSTO**

En el estudio realizado por el Dr. Falkingham<sup>[29]</sup> se observa que la reconstrucción 3D a partir de imágenes 2D es un proceso que requiere tanto equipos de cómputo especializado como de un alto tiempo de procesamiento para adquirir los resultados esperados. Ese nivel de reconstrucción 3D suele ser adecuado para cierto tipo de aplicaciones, pero para actividades de inspección, en un sector como el aeronáutico, que requiere tomar decisiones de una manera rápida y eficiente, resulta una alternativa poco factible. Por esta razón se decidió enfocar la investigación al uso de imágenes para la creación de una representación 3D del escenario inspeccionado de manera automática, evitando la necesidad de que un operador experto sea el que realice la inspección visual en el lugar donde se encuentre la turbina. Se descartó de igual forma

el uso de tecnología láser pues incrementaba la cantidad de componentes que requeriría el robot para realizar la inspección.

Considerando los 5 pasos requeridos para la representación en 3D mostrados en la Figura 19 de la sección 2.3.3, es posible hacer una simplificación de éstos haciendo uso de mallas denominadas primitivas. Al usar estas mallas se simplifica el proceso al enfocarse en la adquisición de imágenes y texturización de la malla primitiva. A continuación, se describen las actividades a realizar para la representación en 3D del escenario de inspección en base a los pasos descritos en la Figura 19 de la sección 2.3.3.

### **3.1.1 Adquisición de imágenes**

La tarea para la adquisición de imágenes de la aplicación final no se encuentra dentro del alcance de esta investigación. Sin embargo, es importante mencionar algunas consideraciones y técnicas necesarias para obtener los mejores resultados posibles con la solución propuesta. En el caso de una inspección boroscópica es posible observar tres actores principales:

- a. Cámara
- b. Operador
- c. Escenario por inspeccionar

El operador es el encargado de desplazar la cámara a lo largo del escenario que desea inspeccionar. Para evitar la necesidad de un operador, el desplazamiento de la cámara se llevará a cabo por medio de un robot, de esta manera se evitan los problemas descritos en la sección 2.6. En la actualidad se emplean en sectores médicos e industriales cámaras de tamaño muy pequeño. La empresa ScoutCam<sup>[34]</sup> ofrece una micro cámara que es utilizada para la inspección de cavidades y monitoreo de objetos pequeños. Esta cámara cuenta con un diámetro de tan sólo 1.2 mm, cuenta con un campo de visión de 130°, una resolución de 49,280 px y requiere una distancia de 3.5 mm entre la lente y el objeto a inspeccionar (la empresa ofrece la posibilidad de reducir esta característica bajo petición). Tomando en cuenta tanto la representación que se le puede dar a la cámara de combustión como las dimensiones de este tipo de cámaras microscópicas, se decidió analizar diferentes configuraciones para la(s) cámara(s) para

poder adquirir imágenes en los ángulos necesarios para representar el ducto marcado en la Figura 12 de la sección 2.2.4.

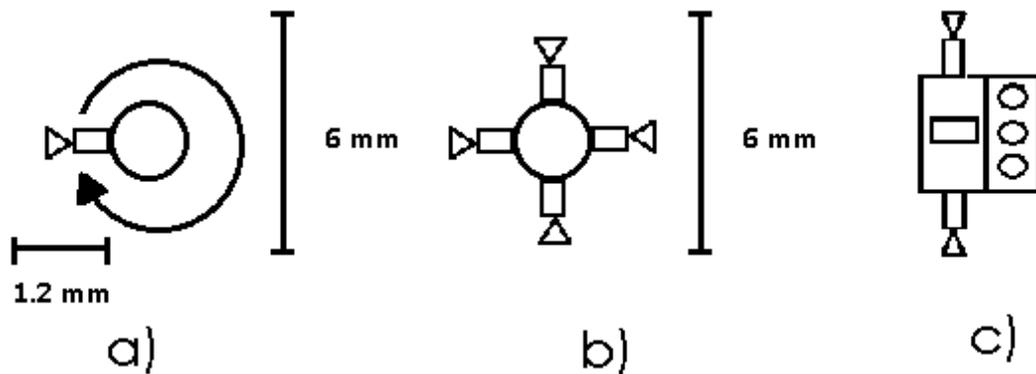


Figura 26. a) Configuración con una sola cámara con un eje rotativo para adquisición de imágenes panorámicas. b) Configuración de 4 cámaras separadas entre si 90°. c) Integración de anillo de iluminación a la configuración con 4 cámaras.

Las configuraciones mostradas en la Figura 26 definen opciones distintas para las imágenes que serán procesadas. La primer configuración requiere solo de una cámara montada en un mecanismo giratorio de 1 grado de libertad que asegura la captura de los 360 grados del escenario, sin necesidad de considerar la distancia que exista entre el robot u objeto portador de la cámara ante alguno de los bordes.

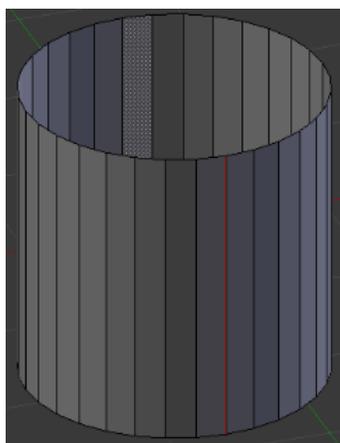
La segunda configuración implica contar con múltiples cámaras en posiciones fijas. La empresa ScoutCam ofrece la posibilidad de solicitar configuraciones a medida para simplificar la integración con otros sistemas dependiendo de la aplicación. Considerando las especificaciones de la microScoutCam 1.2 de la empresa ScoutCam<sup>[50]</sup>, cada una de estas cámaras tiene un ángulo de visión de 130° por lo que esto implicaría que existiría una zona de superposición de 20° entre las cámaras contiguas. Para lograr una correcta captura del escenario, es necesario posicionar el robot u objeto portador de las cámaras en un punto donde sea posible alcanzar el ángulo de visión de 130°; esto con el objeto de procesar correctamente las imágenes y lograr la recreación de una imagen panorámica o de 360°.

Es importante resaltar que a diferencia de una inspección boroscópica, donde la cámara suele adquirir imágenes frontales de donde se encuentra en ese momento, la alternativa propuesta por esta investigación adquiriría imágenes laterales del espacio

inspeccionado. Al tener imágenes en 360 grados es posible tener una mejor visión del detalle de las diferentes paredes que conforman el escenario conforme se va realizando la inspección.

### **3.1.2 Definición de malla para utilizar en inspección boroscópica**

La inspección boroscópica consiste en el desplazamiento de una cámara a lo largo de un espacio o escenario. Al considerar el escenario a reconstruir, es posible utilizar una malla primitiva cilíndrica (ver Figura 27). Considerando las configuraciones recomendadas en la Figura 26, se obtendría una imagen o conjunto de imágenes que terminarían recreando el escenario en 360°.



*Figura 27. Malla cilíndrica generada en Blender.*

A pesar de que el escenario en la realidad tuviese una forma distinta, pudiendo ser representado un poco mejor por un rectángulo, se decidió hacer uso de la malla cilíndrica pues se considera una mejor manera para representar el tipo de imagen adquirida. La solución que se describirá posteriormente hace este uso de malla. Sin embargo, es posible realizar pequeñas modificaciones una vez que se tengan imágenes adquiridas ya con el prototipo del sistema de inspección.

### **3.1.3 Texturización de la malla**

Para concluir con la representación propuesta del escenario en tres dimensiones que podrá ser utilizado por el operador para su análisis, solo nos faltaría realizar la texturización

de la malla. Para realizar esto requerimos desenvolver el cilindro para tener una proyección en un plano como se muestra en la Figura 28.

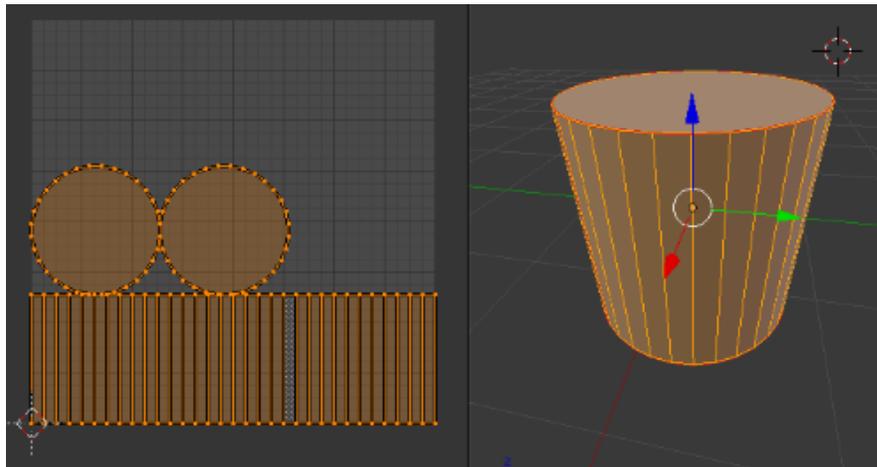


Figura 28. Función 'Unwrap' en Blender de una malla cilíndrica.

La proyección plana de un cilindro nos permite posicionar en caras específicas una o más imágenes para su texturización.

## 3.2 REPRESENTACIÓN EN 3D DEL ESCENARIO

Para evitar la necesidad de un operador durante la inspección se requiere que el sistema capture y despliegue de manera clara el estado interno del componente inspeccionado. Para lograr esto la representación creada debe ser similar a la que obtenida en una inspección manual. La ventaja de una representación 3D es que le permite al operador visualizar en 360° el modelo, permitiéndole prácticamente recrear la inspección.

### 3.2.1 Malla cilíndrica

La malla definida para esta aplicación fue la cilíndrica, considerando que el escenario puede representarse como un túnel, existen dos formas para lograrlo; esto a través de Blender y en este tipo de malla. Por un lado, la Figura 29 representa el escenario como un solo cilindro seccionado en  $n$  partes. Cada parte tendría una correspondencia directa con un punto específico del escenario.

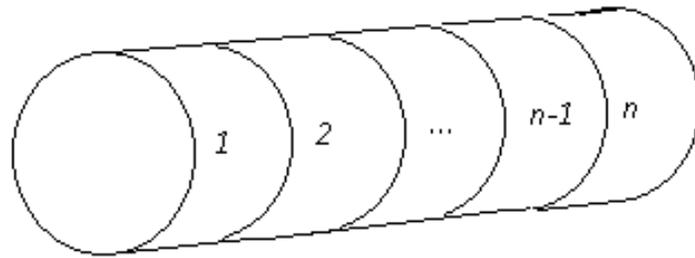


Figura 29. Una sola malla cilíndrica con  $n$  cortes para representar cada sección del escenario.

Otra alternativa es considerar cada sección como una malla independiente como se representa en la Figura 30. De esta manera en lugar de contar con una malla seccionada, sería seccionar el escenario y representar cada una de estas divisiones en una malla independiente. Al final, la agrupación de las  $n$  mallas nos daría la representación completa de nuestro escenario.

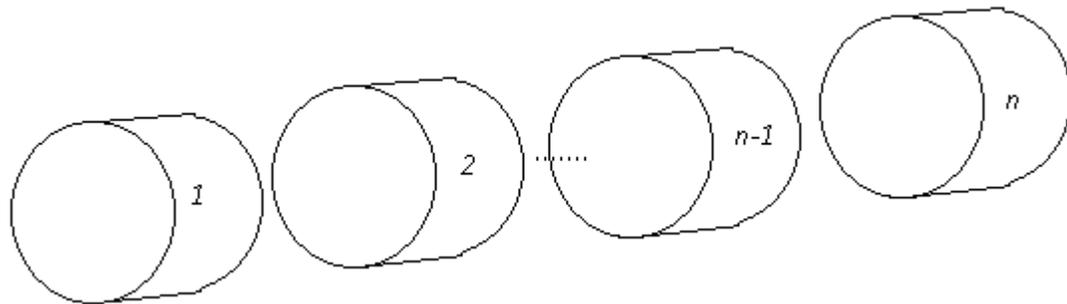


Figura 30. Múltiples mallas que representan una sección del escenario fragmentado.

Por la simplicidad, tanto en la calibración como en el procesamiento de las imágenes para texturizar el modelo, se consideró una mejor opción el trabajar con múltiples mallas. De esta forma, el robot suave debe ser configurado para la correcta adquisición de imágenes. Esta configuración debe considerar la velocidad de desplazamiento del robot, el tiempo de captura de la imagen y la distancia a recorrer antes de adquirir las imágenes correspondientes a la próxima malla. De esta forma es posible representar el escenario a través de la unión de varios conjuntos, para los cuales cada conjunto de imágenes corresponde a una malla de todo el escenario inspeccionado.

### 3.2.2 Script en Blender

A continuación, se describe paso a paso las diferentes secciones del funcionamiento del script de Python utilizado en Blender para la recreación del escenario. Cabe resaltar que

la interfaz gráfica de Blender muestra al usuario los comandos de su API en Python relacionados cada una de las acciones. De esta manera el desarrollo del script fue más sencillo pues se trabajó en una primera etapa de manera manual (creando mallas, texturas y materiales, así como la generación del modelo) siendo más sencillo realizar la depuración de errores, para hacer la traducción y lograr la automatización de principio a fin de la representación del escenario. La Figura 31 muestra el diagrama de alto nivel sobre el cual se basa el script creado.

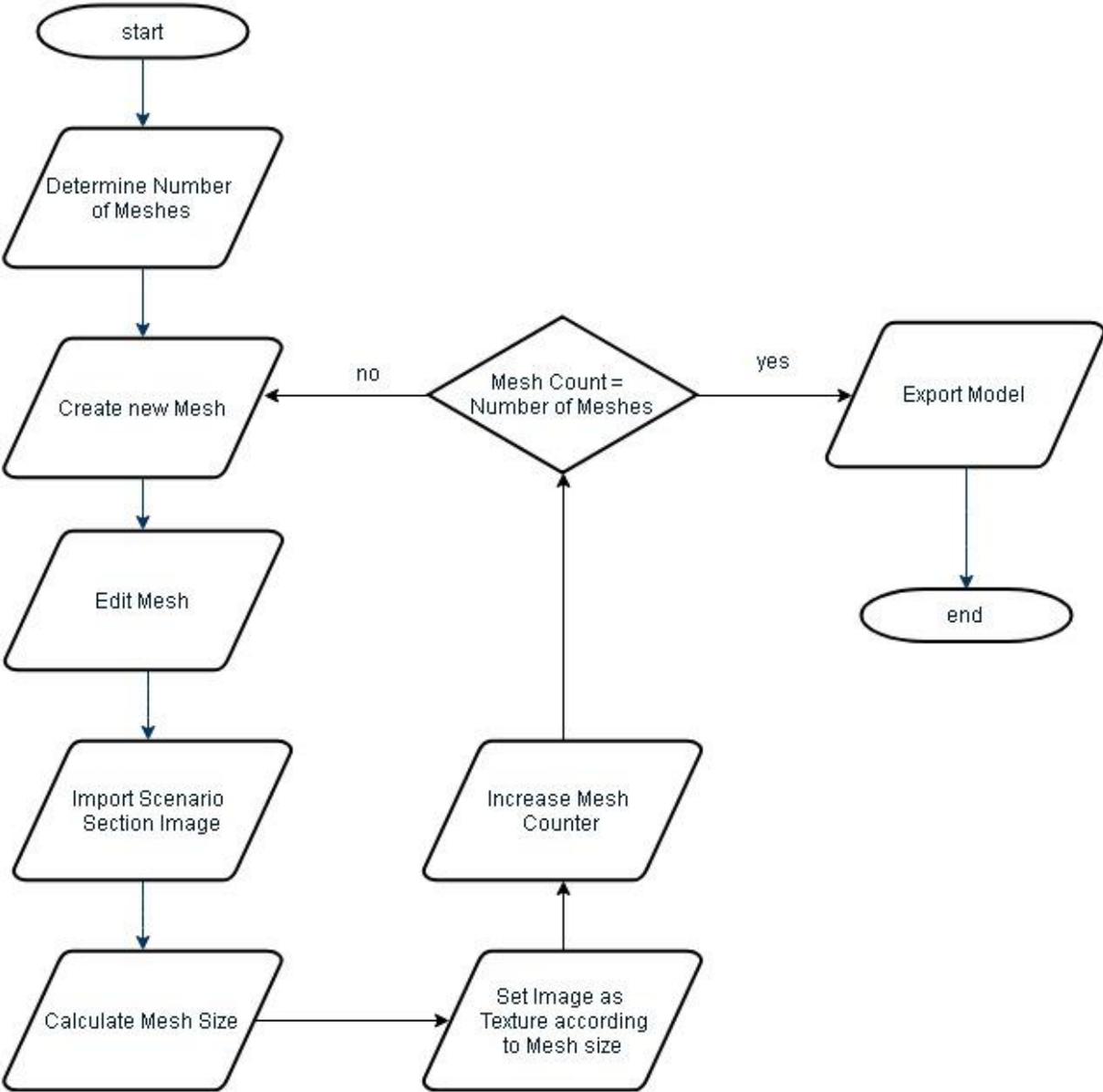


Figura 31. Diagrama de flujo de alto nivel que muestra el funcionamiento del script de Blender.

Este script basa su funcionamiento en el trabajo de imágenes panorámicas. Cualquier procesamiento necesario para adquirir este tipo de imágenes debe ser realizado previamente. El script define en la línea 12 un folder local de donde leerá cada imagen panorámica, las cuales deben ser una representación de cada sección del escenario a inspeccionar, por lo que el número total de imágenes será el número total de mallas requeridas para formar el modelo. La Figura 32 muestra el código fuente donde se define el folder que contendrá las imágenes y donde se calcula el número de imágenes encontradas para determinar la cantidad de mallas.

```
1 import bpy
2 import bmesh
3 import os
4
5 #bpy.ops.object.mode_set(mode='OBJECT') #set object mode
6 bpy.ops.object.delete(use_global=False)
7 def ls(cdir): # return dir contents of cdir
8     ls = os.listdir(cdir)
9     return ls
10
11 sc = bpy.data.scenes[0] # get current scene
12
imgdir="C:\\Users\\Maduch\\Documents\\Maestria_SIM\\Tesis\\Imagenes\\Test\\Test6"
13 img_ext='jpg'
14
15 files=[] # list for files
16 images=[] # list for image files
17 files=(ls(imgdir)) # read file list into list
18 file_count=len(files) # file count
19
20 for a in range(0,file_count): # for each file:
21     if files[a].endswith(img_ext): # does it end with?
22         images.append(files[a]) # if so then add to images list
23
24 bpy.ops.file.autopack_toggle()
25 image_count=len(images) # count of images
26 print(file_count)
27 print(image_count)
28 pos = -6 #initial position of first mesh
29 # (needs to be adjusted for real
30
```

Figura 32. Dentro de la configuración inicial se encuentra el folder donde se localizan las imágenes. De igual manera, se define la posición donde se encontrará la primera malla.

Una vez determinada la cantidad de mallas requeridas para la recreación del escenario se comienza a generar cada una de las mismas. Dependiendo de la aplicación que se desee utilizar, es posible modificar el tipo de malla primitiva. La investigación actual considera una malla cilíndrica, esta malla posee diferentes campos configurables relacionados con su tamaño, posición y rotación. Para cuestiones de visualización, y

considerando que la inspección boroscópica puede ejemplificarse como el desplazamiento a lo largo de un túnel, se ha configurado la malla con una rotación de 90° y eliminando el relleno de la malla para solo trabajar con los contornos. El tamaño del cilindro puede ser modificado por el parámetro *radius*. El código fuente presentado en la Figura 33 muestra en la línea 39 la configuración de estos parámetros. En el caso de la rotación de 90 grados, el valor utilizado es 1.57 que representa los radianes de rotación. En la Figura 34 se presenta como se visualizaría la malla cilíndrica previamente definida.

```
31 for a in range(0,image_count): # for each image
32     print("=====")
33     print('loop count: '+str(a))
34     # CREATE MESH
35     # end_fill_type = NOTHING to make a hollow cylinder
36     # radius corresponds to define the radius size of the cylinder
37     # location(x,y,z) location on y to be increased for each new panorama
38     # rotation 90 degrees in order to make a tunnel form
39     bpy.ops.mesh.primitive_cylinder_add(end_fill_type='NOTHING',
radius=2,location=(0,pos,0), rotation=(1.57,0,0))
```

Figura 33. Código para la creación y configuración de la malla cilíndrica.

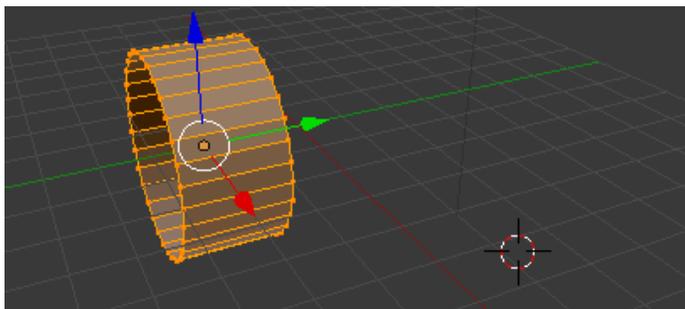


Figura 34. Visualización de la primera malla en Blender.

Una vez creada la malla es necesario configurar algunas propiedades. La Figura 35 muestra el código fuente requerido para la texturización de la malla. En primer lugar, es necesario indicar que nos interesa realizar la texturización de la malla en su interior y no solamente en el exterior, esto se logra con la función *flip\_normals()* llamada en la línea 46. De igual forma, debemos indicar la forma en cómo nos interesa realizar esta texturización. En este caso queremos que todo el interior de la malla sea texturizado, por lo cual se hace uso del comando *bpy.ops.mesh.select\_all)* como se muestra en la línea 63.

```

40
41 bpy.ops.object.mode_set(mode='OBJECT', toggle=False) #set context to
object mode
42 bpy.ops.object.shade_smooth() #smooth property displays the mesh
without showing the edges
43 plane=bpy.context.scene.objects.active
44 bpy.ops.object.mode_set(mode='EDIT') #set edit mode to
perform texturization
45
46 bpy.ops.mesh.flip_normals() #flip normal to fill mesh inside
47
48 # force edge selection mode
49 bpy.context.tool_settings.mesh_select_mode = (False, True, False)
50
51 # this will select every 'even' numbered edge from the list of edges.
52 obj = bpy.context.active_object
53
54 bm = bmesh.from_edit_mesh(obj.data)
55 for edge in bm.edges:
56     if edge.index == 66:
57         edge.select = True
58         continue
59
60     edge.select = False
61
62 # Definition of how the mesh is going to be unwrapped
63 bpy.ops.mesh.mark_seam(False)
64 bpy.ops.mesh.select_all(action='SELECT')
65 bpy.ops.uv.unwrap()
66

```

Figura 35. Código de preparación de la malla para realizar la texturización.

Dentro de las propiedades de la malla, se encuentra la opción de sombreado. Esta opción habilita o deshabilita la visualización de los vértices a lo largo de la malla. Para una mejor visualización del modelo, con opción *smooth* es posible esconder dichos vértices para que se vea de una manera más uniforme la malla. En la Figura 36 se observa como la función *smooth* le da una apariencia más lisa a nuestra malla.

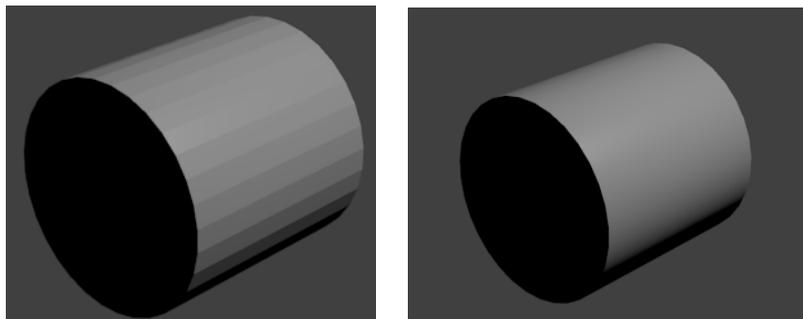


Figura 36. A la izquierda la configuración flat de la malla. A la derecha la configuración smooth.

Posteriormente, es necesario agregar la textura. En esta parte se debe importar la imagen que se utilizará para la misma y plasmarla a lo largo de la malla extendida. Se debe ajustar la zona de la malla a las dimensiones de la imagen. Para esto se hace uso de la líneas 68 y 69 del código mostradas en la Figura 37. La transformación *translate* desplazará la malla de manera vertical para lograr centrarla con respecto a la imagen. La transformación *resize* sirve para cambiar el tamaño a la proporción adecuada de la malla. La Figura 38 muestra el resultado obtenido, en el cual se observa como la malla abarca en su totalidad la imagen con la cual se desea texturizar. Al estar trabajando con imágenes panorámicas, estos ajustes se aplicarán sobre el plano y. Lo anterior se debe a que, al momento de incorporación de la imagen, el eje x es posible ajustarlo automáticamente con la propiedad *clip*, ejecutada en la línea 80, utilizado en las propiedades de la textura.

```
66
67 # perform transformation fot better display results
68 bpy.ops.transform.translate(value=(0.0, 0.4, 0.0)) #0.14
69 bpy.ops.transform.resize(value=(1.0, img.size[1]/22.5, 1.0)) #/1300
70
71 bpy.context.area.type = last_area
72
73 mat=bpy.data.materials.new('mat'+str(a)) #Define Material name
for texture
74 bpy.context.object.data.materials.append(mat)
75
76 tex=bpy.data.textures.new('ColorTex', type = 'IMAGE')
77 imgpath=imgdir+'\\'+images[a] # make string with path ti image
78 img = bpy.data.images.load(imgpath) # load image
79 tex.image = img
80 tex.extension = 'CLIP'
81 mtex = mat.texture_slots.add()
82 mtex.texture = tex # add texture
83 mtex.texture_coords = 'UV' #change to OBJECT and select Cylinder
84 mtex.mapping= 'FLAT'
85 imgX = img.size[0]/1000.0 # calculate dimensions
86 imgY = img.size[1]/1000.0
87
88 pos = pos + 2
89 bpy.ops.object.mode_set(mode='OBJECT') #set object mode
90
```

Figura 37. Código para la adición de textura a la malla.

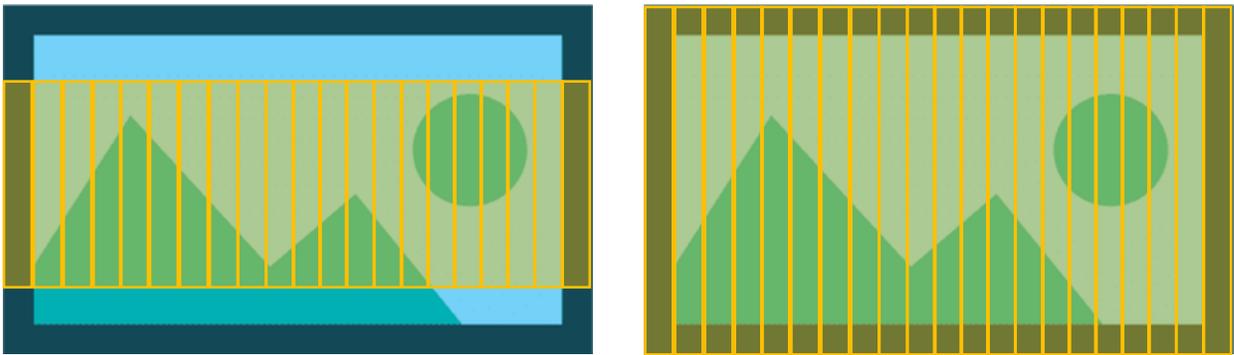


Figura 38. A la izquierda mapa de texturización antes de realizar transformaciones. A la derecha, mapa de texturización después de realizar las transformaciones.

Finalmente, el script continuará ejecutando estos procesos hasta agregar la malla correspondiente a la última sección del escenario. En la Figura 39 es posible apreciar un modelo creado con un conjunto de 3 mallas, la región naranja remarca la última malla creada. Al final de este proceso, se podrá apreciar un modelo parecido a una tubería, y por el cual el operador podrá desplazarse, ejemplificando de esta manera la operación que actualmente realizan al estar moviendo la cámara boroscópica. Este modelo le permite al operador moverse a lo largo del escenario, así como rotar la vista de acuerdo con el área que se quiera observar con detenimiento.

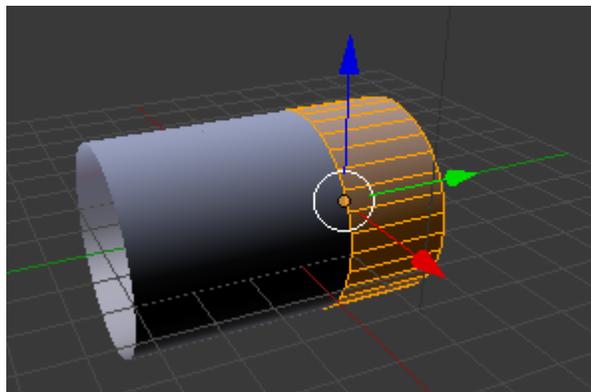


Figura 39. Ejemplo de adición de nueva malla (sección).

Para la visualización del modelo, solamente realizamos la exportación directamente de Blender a Sketchfab, y una vez exportado, es posible navegar por la página donde se encuentra el modelo. El script descrito en esta sección está basado en el uso de imágenes panorámicas para su funcionamiento. Como se mencionó en la sección 3.1.1, esta investigación se enfocó a dos posibles configuraciones del sistema que adquirirá las imágenes. Aunque pareciera que esta solución se enfoca a la primera configuración, la cual consiste en una sola cámara localizada en una base rotativa para la adquisición

de una imagen panorámica, es posible expandir esta solución a la segunda configuración que consta de cuatro cámaras fijas a una separación de 90° entre ellas. Para lograr esto, es necesario convertir las imágenes adquiridas por cada cámara a un panorama.

### 3.3 AGRUPACIÓN DE IMÁGENES

Para esta tarea se consideró el uso de la librería de Python OpenCV, la cual está enfocada a aplicaciones de visión computarizada. Esta librería posee ya funciones predefinidas para la realización del *stitching* entre dos o más imágenes. Donato<sup>[36]</sup> describe el uso de un script creado por Adrian Rosebrock haciendo uso de OpenCV y Python. Este script consiste principalmente en posicionar en un directorio la serie de imágenes que se desean juntar y el *stitching* se realiza a través de la función *createStitcher* de la clase *Stitcher* de OpenCV.

Este algoritmo ofrece la robustez de realizar la agrupación de las imágenes sin necesidad de que estas mismas estén ordenadas. De igual forma, es capaz de detectar correctamente las zonas comunes con poca variación de iluminación, la cual puede obtenerse por el ángulo de captura de la cámara. Para que funcione adecuadamente, es necesario que las imágenes tengan regiones fáciles de diferenciar y que además sean compartidas entre dos o más imágenes para que se pueda identificar el punto de superposición<sup>[36]</sup>. Una de las limitaciones que tiene este algoritmo es que las imágenes deben estar alineadas horizontalmente. Para agrupación de imágenes de manera vertical, es necesario rotar cada imagen 90 grados, creando así una alineación horizontal, ejecutar el algoritmo para que se realice la agrupación de imágenes y por último rotar la imagen resultante -90 grados para que se despliegue de manera vertical.

La cámara de combustión es un componente que posee diferentes características como los orificios de presión de aire e inyectores de combustible con las cuales se puede identificar estas regiones compartidas. Normalmente, los inyectores de combustible se localizan en la pared con dirección hacia el centro de la turbina. Las paredes continuas a los inyectores tendrán los orificios de presión de aire, mientras que la pared restante generalmente consiste en unas rejillas de ventilación. La Figura 40 muestra estos componentes y como cada uno se localiza en una pared distinta, a excepción de los

orificios de presión que se localizan en paredes opuestas entre sí. Para este componente, estas características deben ser suficientes porque se diferencian entre si los componentes de cada una de sus caras. Igualmente, considerando que la cámara propuesta tiene un ángulo de visión de 120°, debe detectarse una cara completa de ducto de combustión junto con partes de las áreas contiguas, facilitando la identificación de zonas comunes.



Figura 40. Imagen del interior de la cámara de combustión. La figura de la izquierda muestra la pared de inyectores y los orificios de presión. La figura de la derecha muestra la rejilla de ventilación entre las dos paredes laterales con orificios de presión<sup>[51]</sup>.

### 3.4 EJECUCIÓN DE PRUEBAS

Para la ejecución de pruebas se escogieron 3 escenarios con fácil acceso desde su interior, simulando el punto de adquisición de imágenes donde se localizaría el robot. De cada escenario se obtuvieron imágenes panorámicas, las cuales fueron utilizadas posteriormente para la creación de una representación 3D con base en la descripción de la Figura 31. Para este tipo de pruebas solo es necesario colocar las imágenes panorámicas en un folder y actualizar el script de Python descrito en el Anexo A, sustituyendo la ubicación del folder. Debido al nivel de profundidad de los escenarios, se decidió utilizar la misma imagen varias veces para darle volumen al modelo obtenido.

Posterior a la adquisición de imágenes panorámicas, se realizaron pruebas adquiriendo 4 imágenes de los últimos dos escenarios inspeccionados. Una vez obtenidas las imágenes se almacenaron en un folder y se ejecutó el script de la función de *stitching* que se encuentra integrada con OpenCV<sup>[36]</sup>. Una vez obtenida la imagen panorámica a partir de las 4 imágenes laterales del escenario, se usó nuevamente el *script* de Python del anexo A el cual ejecuta los pasos mencionados en la Figura 31 en la sección 3.2.2,

obteniendo de esta forma la representación del escenario utilizando mallas primitivas cilíndricas. El anexo B es un script equivalente que genera la representación del escenario con mallas primitivas rectangulares, el cual fue utilizado para mostrar las diferencias con el primer tipo de malla y listar las consideraciones adicionales requeridas para este tipo de malla.

## 4. RESULTADOS

En este capítulo se describen los resultados obtenidos a través del script en Python para la representación del escenario en 3D. Para las pruebas realizadas se utilizaron imágenes adquiridas en espacios cerrados. No fue posible adquirir imágenes de previas inspecciones boroscópicas por cuestiones de confidencialidad, así como también debido a que este tipo de inspecciones suelen utilizar una cámara con punto de enfoque frontal y no lateral, dificultando de esta forma poder adquirir una imagen panorámica o las múltiples imágenes a 90° de posicionamiento. A pesar de esto, los resultados obtenidos con las imágenes adquiridas en espacios cerrados muestran la escalabilidad y variedad de aplicaciones que pueden llegar a ser beneficiadas con esta investigación.

### 4.1 RESULTADOS CON FOTOS PANORÁMICAS

El uso de fotos panorámicas resultó bastante aceptable. Un punto a mencionar es que el script creado en Python para la representación del escenario en Blender, requiere la definición del punto inicial, es decir, donde comenzará a insertarse la imagen como textura en la malla del modelo. Conociendo esto, es importante realizar la captura de la imagen panorámica iniciando desde el mismo punto, para que de esta forma se logre visualizar correctamente. La Figura 41 muestra una imagen panorámica tomada en el interior de un elevador para su posterior procesamiento.



*Figura 41. Imagen panorámica adquirida dentro de un elevador.*

La generación del modelo tomó 10 segundos utilizando una computadora con 8GB de RAM, Intel Core I3 de cuarta generación y una tarjeta gráfica Intel® HD con 1GB de memoria de video. Para este modelo se utilizaron 5 copias de la imagen panorámica con un tamaño de 3.2 MB, de alta resolución. El modelo generado tuvo un tamaño de casi 33 MB. En la Figura 42 se muestra tanto el modelo creado automáticamente en

Blender, como su representación en 3D que puede ser utilizada para el proceso de valoración por parte de un inspector. Dicho modelo fue publicado a través de Blender a Sketchup para su visualización. La imagen utilizada fue probada con una resolución menor, obteniendo resultados parecidos pero el nivel de detalle en la vista interior resultó de menor calidad.

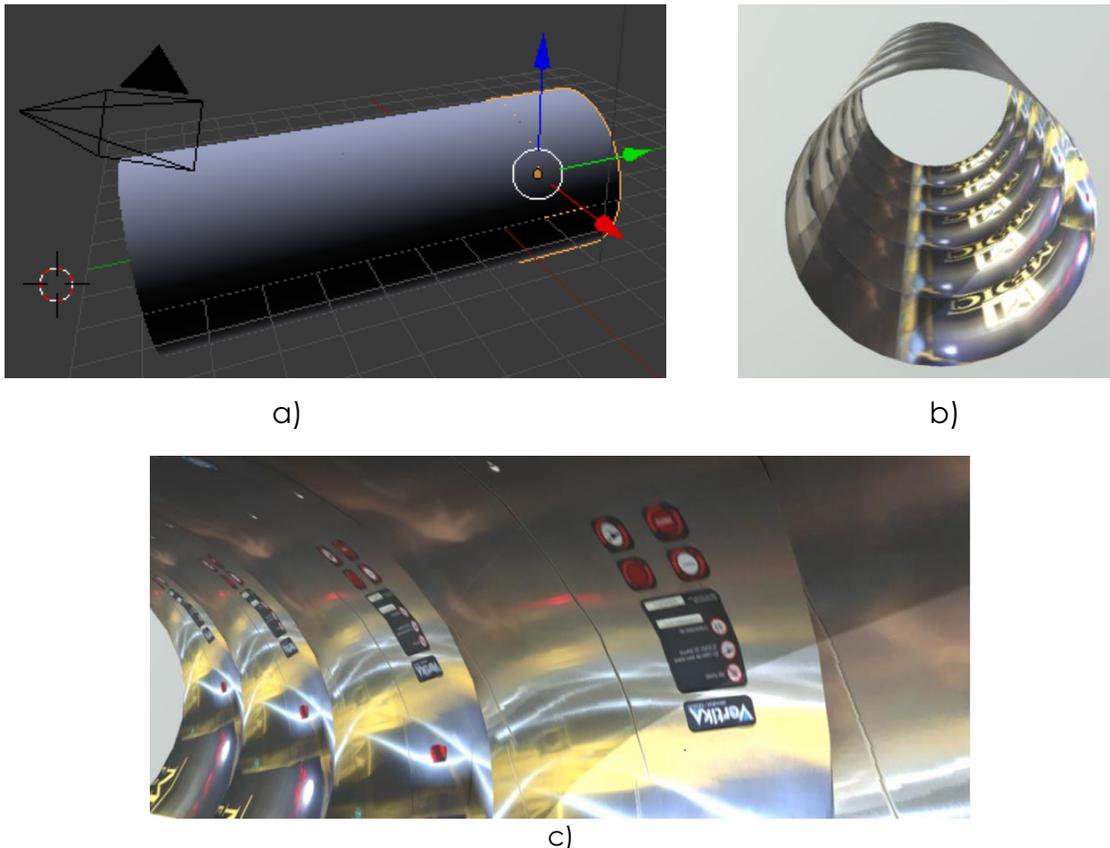


Figura 42. a) Modelo generado en Blender conformado por 5 mallas cilíndricas. b) Vista frontal del modelo en Sketchfab. c) Vista interior del modelo 3D en Sketchfab.

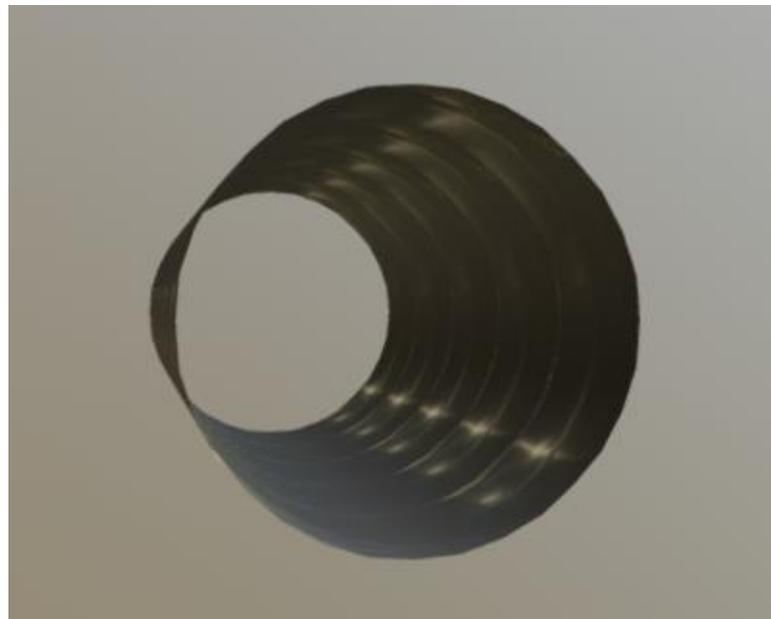
Las Figuras Figura 43 y Figura 44 replican el proceso anterior, pero en esta ocasión adquiriendo una imagen panorámica desde el interior de un ducto de ventilación, algo más semejante a lo que se esperaría de las imágenes de una cámara de combustión. En este caso es posible observar que la oscuridad dentro del ducto es un factor que podría dificultar la correcta interpretación del escenario. Es por esta razón que se recomienda que el dispositivo final cuente con un sistema de iluminación que ayude a la adquisición clara de las imágenes, pues la cámara de combustión se encuentra en una región con poca luz natural. La compañía ScoutCam<sup>[50]</sup> ofrece algunas alternativas de integrar un conjunto de iluminación a las cámaras con el cual se podría abatir este problema. Otra alternativa sería la de añadir un anillo de luz led al dispositivo en lugar de

a cada cámara. Ambas opciones anteriores deben ser estudiadas considerando la limitante del punto de acceso a la región de la cámara de combustión. Sin embargo, se sugiere optar en primera instancia por un anillo de luz localizado antes de las cámaras para que, de esta forma, la iluminación otorgada a la escena a inspeccionar sea compartida por todas las cámaras creando una iluminación uniforme.

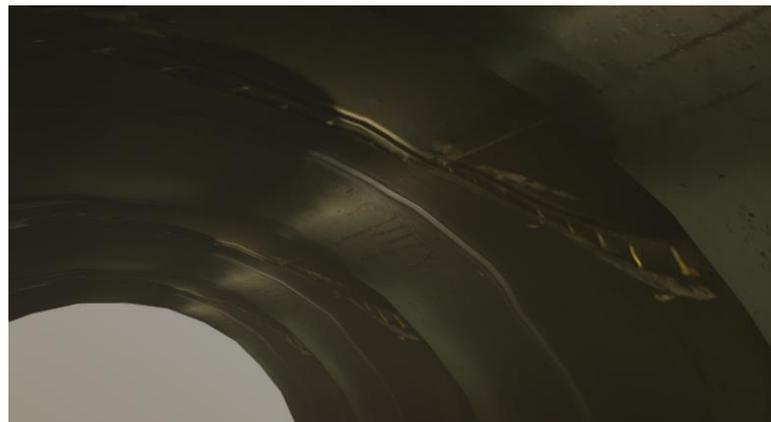


Figura 43. Imagen panorámica del interior de un ducto de ventilación.

a)



b)



c)

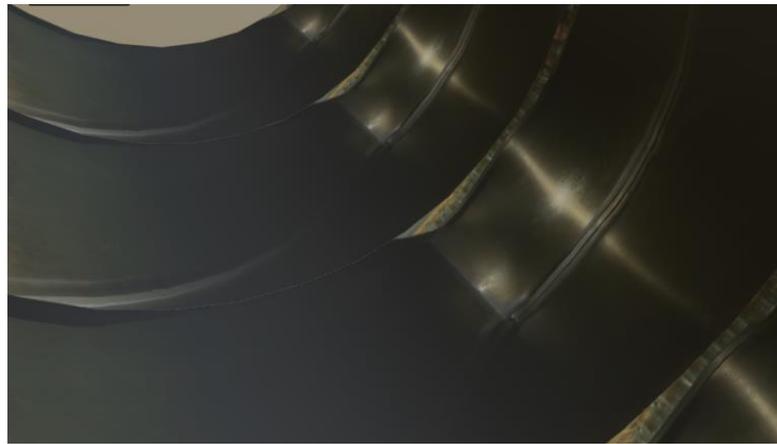


Figura 44. a) Representación del modelo generado. b) Fotografía interna del lado superior del modelo. c) Fotografía interna del lado inferior derecho del modelo.

## 4.2 RESULTADO CON MÚLTIPLES FOTOS

Se realizaron pruebas de creación panorámica a partir de múltiples imágenes. En la Figura 45 se presenta la adquisición desde cuatro puntos distintos del ducto de ventilación previamente capturado en una imagen panorámica. Los resultados no fueron como se esperaba. Mientras que la librería de OpenCV provee resultados bastante aceptables en la creación de panorámicas con imágenes de ambientes exteriores con buena iluminación, como lo muestra Donato<sup>[36]</sup>, la poca iluminación del escenario representó un problema pues no se lograba identificar las regiones comunes para la correcta agrupación de las imágenes. Realizando la depuración de los resultados, la proyección de la luz es un factor muy importante que considerar, de igual forma el ducto capturado con estructura metálica, mostraba mucho brillo con la iluminación por lo que el script no fue capaz de identificar correctamente las regiones comunes entre las imágenes y debido a esto la operación de *create\_Stitcher* fue fallida. Considerando las propuestas realizadas en la sección 4.1, referentes a la adición de un conjunto de iluminación similar al utilizado en las inspecciones actuales, este tipo de problemas debido a la poca iluminación no se presentarán al momento de que el robot realice la inspección visual de la cámara de combustión.

Finalmente, se encontró que al tomar las imágenes con una sola cámara cambiando el ángulo para lograr adquirir 4 imágenes que capturarán los 360° del ducto, algunas de ellas presentaron una inclinación en el eje Z al no estar correctamente fija la cámara. Por lo que esta inclinación afectó en la búsqueda de regiones comunes, generando errores

al momento de procesar las imágenes como se puede observar en la Figura 46. Al contar con un eje fijo donde se puedan posicionar las diferentes cámaras será posible minimizar el error cuando ocurra.

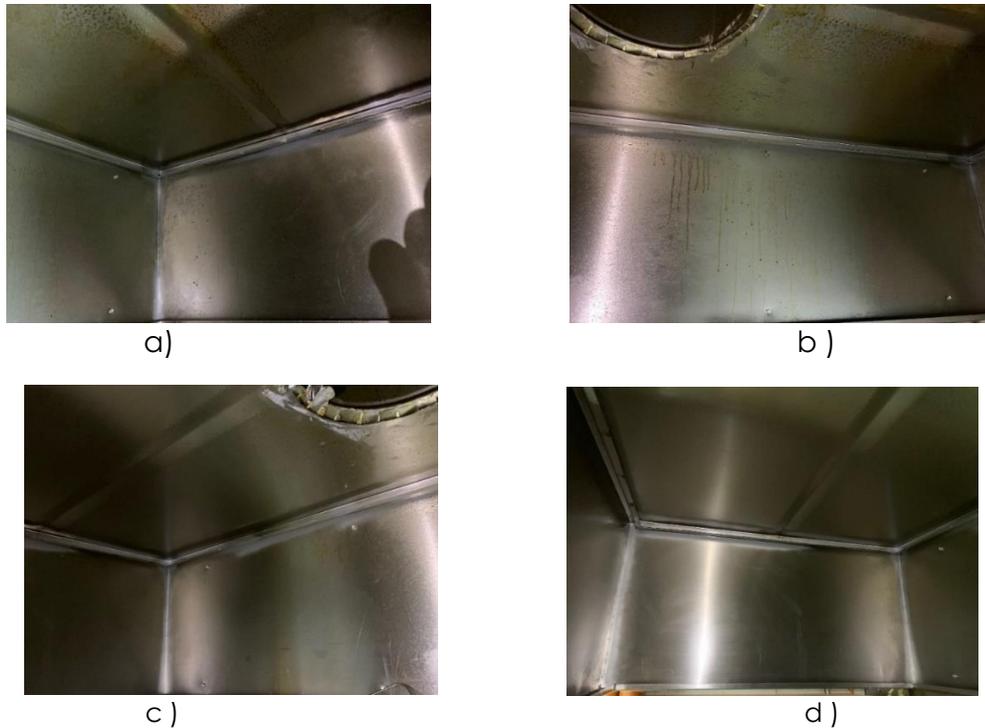


Figura 45. Imágenes capturadas a distintos ángulos dentro de un ducto de aire.

```
[INFO] loading images...  
[INFO] stitching images...  
[INFO] image stitching failed (1)
```

Figura 46. Mensaje de error de la función `create_Stitcher` de Python.

Se decidió probar el software de *Fiji*<sup>[53]</sup> con las imágenes capturadas del ducto de aire. Primeramente, este software fue capaz de otorgarnos una imagen resultado a diferencia de OpenCV. En segundo lugar, la configuración utilizada para el *Stitch* fue por el nombre de la imagen. Al ser imágenes adquiridas secuencialmente, era posible identificar la secuencia de las imágenes a través de su nombre. En la Figura 47 es posible observar los resultados obtenidos con este software. Los resultados obtenidos no fueron aceptables pues la agrupación de imágenes no fue lograda correctamente. Es importante mencionar que la adquisición de estas imágenes fue manual sin ningún tipo de soporte,

por lo que es posible que se puedan obtener mejores resultados teniendo un soporte fijo y posicionando cada cámara en los ángulos correctos.

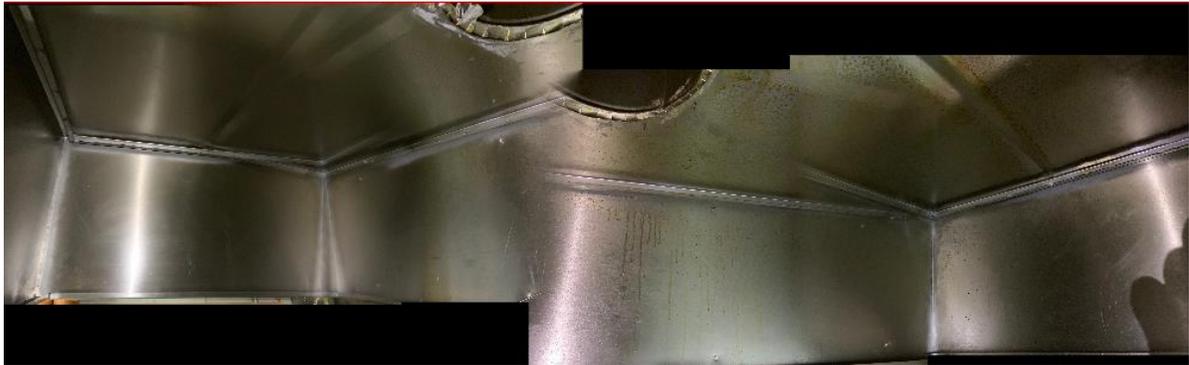


Figura 47. Imágenes del ducto de aire producido con Fiji.

La inclinación mencionada anteriormente es posible percibirla en los espacios negros del resultado por Fiji<sup>[53]</sup>. Principalmente, esta inclinación afecta en la identificación de los bordes donde el ducto muestra un doblez vertical. Se decidió realizar una tercera prueba en un objeto donde fuera más sencillo mantener la cámara fija para adquirir las cuatro imágenes, es decir, sin esta inclinación. Se decidió capturar el interior de un horno de una estufa por su fácil acceso para la adquisición de imágenes; la Figura 48 es un bosquejo de las dimensiones del mismo. Cabe mencionar que se utilizó la iluminación interna de la estufa y las imágenes adquiridas fueron sin flash, para de esta forma evitar resultados no deseados debido al reflejo del flash de la cámara en la superficie de la estufa. Una vez capturadas, se utilizó la librería de OpenCV con la función *create\_stitcher* y se compararon los resultados con una imagen panorámica capturada con la misma cámara. En esta ocasión al mantener la inclinación en el eje Z lo más cercano para todas las fotos adquiridas, los resultados obtenidos fueron más aceptables que en las pruebas anteriores.

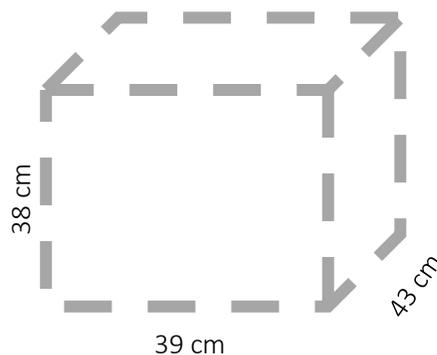


Figura 48. Dimensiones del horno inspeccionado.

El escenario a inspeccionar tiene un perímetro de 154 cm considerando sus cuatro caras por 43 cm de fondo. Cada una de las imágenes adquiridas utilizadas para la creación de una imagen panorámica presenta unas dimensiones de 1400x728 pixeles, analizando los pixeles por pulgada (ppp) en la información de cada imagen, se pudo determinar que todas las imágenes fueron capturadas con una resolución de 72 ppp. Dividiendo el número de pixeles entre la resolución podemos obtener la cantidad la medición real de la imagen, obteniendo de esta forma que la imagen mide 19.44 x 10.11 pulgadas cuadradas o su equivalente en centímetros cuadrados sería 49.37 x 25.67 cm<sup>2</sup>.

En la Figura 49 se muestran las imágenes capturadas desde distintos ángulos del escenario. Se identificó el centro de cada pared del horno con una etiqueta para poderlas identificar fácilmente al finalizar el proceso. Estas etiquetas son meramente informativas y no tienen ningún efecto en el proceso de *stitching* debido a que se localizan en el centro de la imagen y no se localizan en las regiones laterales en las que se enfoca el algoritmo para la generación de la panorámica.

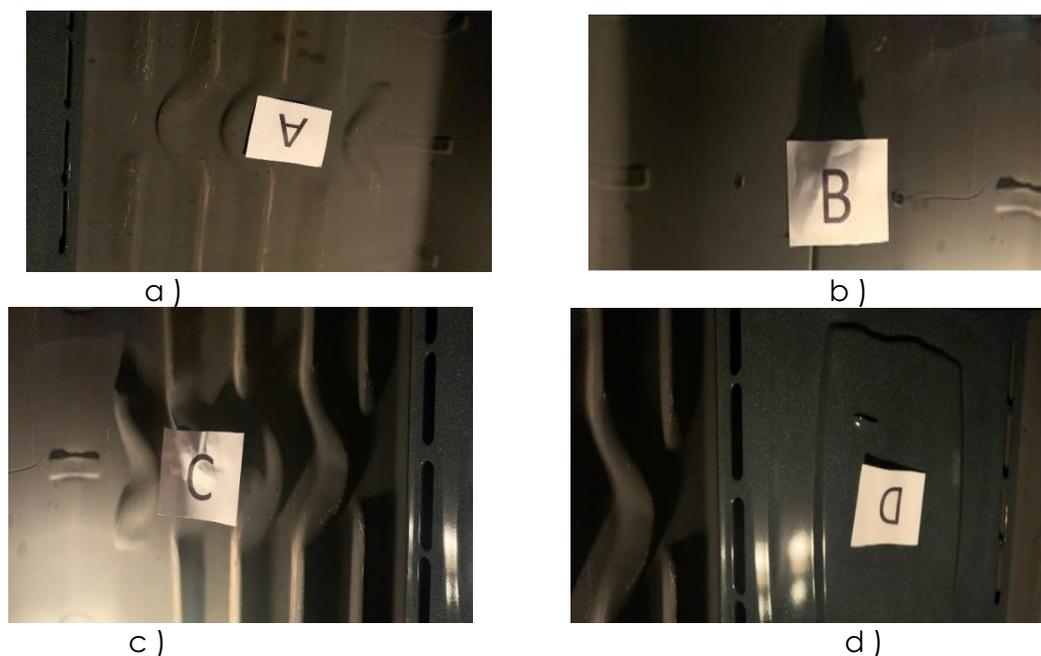


Figura 49. Imágenes capturadas a distintos ángulos dentro del horno de una estufa.

Las imágenes obtenidas una vez realizado el *stitching* presentan un tamaño de 4256x676 pixeles que son equivalentes a 150.14x23.85 cm<sup>2</sup>. De igual forma se editó la imagen obtenida para remover los extremos repetidos, con lo que se obtuvo que la imagen real de todo el escenario tiene un tamaño de 4016x676 pixeles o 141.68x23.85 cm<sup>2</sup>. Con estos datos pareciera que se obtuvo una imagen 12.5 cm más pequeña (considerando el

perímetro únicamente). Sin embargo, se analizaron las diferencias entre las imágenes adquiridas y la imagen creada y se encontró una discrepancia en la cantidad de píxeles obtenidas en las esquinas contra la medición real. Para lograr determinar dicha diferencia, se realizó una medición de una esquina, logrando determinar que la imagen capturada presentó 3 cm menores a los medidos con una cinta métrica. La Figura 50 muestra el área que se midió, donde la medición real era de 14 cm, y los píxeles comprendidos por esa área eran los equivalentes a 11 cm. La diferencia obtenida no se relaciona con pérdida de información sino es más bien un efecto de la transformación a imagen panorámica. Sin embargo, la información necesaria para ser analizada y estudiada por el operador aún sigue siendo parte de la imagen, aunque en una cantidad menor de píxeles. Tomando en cuenta la cantidad de píxeles reducidos, es posible decir que una reducción de entre el 7% y 10% de píxeles en comparación con el tamaño real es aceptable para contar con toda la información del escenario para determinar su estado actual.



*Figura 50. Área analizada para determinar diferencias entre el perímetro real y el perímetro capturado en las imágenes.*

Considerando lo anterior, se puede concluir que la imagen obtenida es bastante cercana a lo esperado, pues sus dimensiones son muy cercanas a las de la imagen panorámica, tan solo con las discrepancias en las esquinas previamente explicadas. En la Figura 51 se pueden observar los resultados obtenidos de la creación de una imagen panorámica a partir de 4 imágenes.

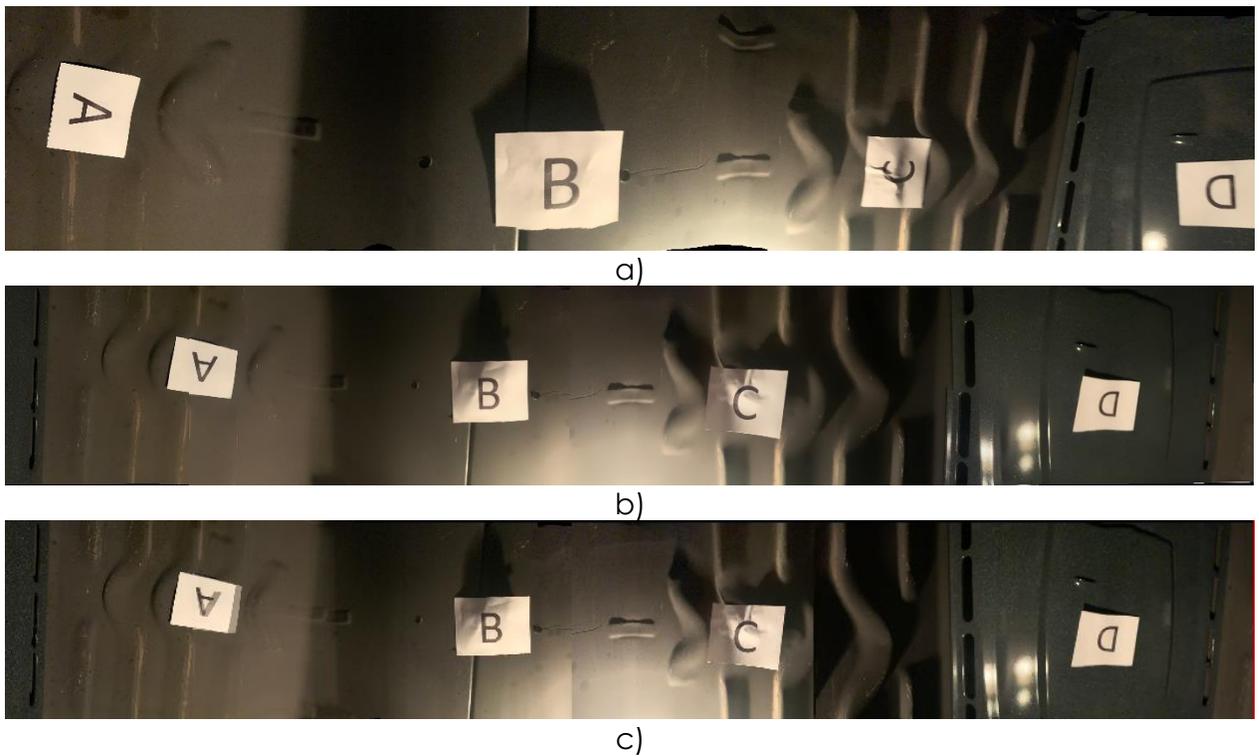
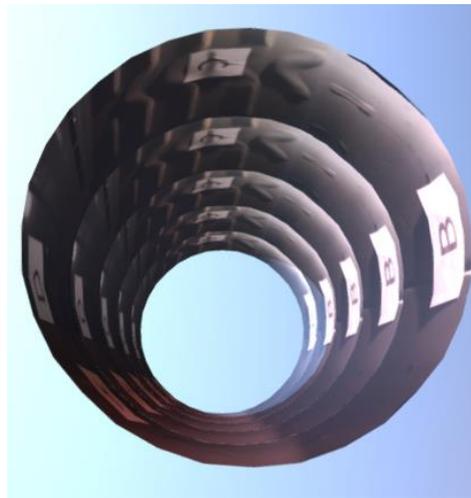


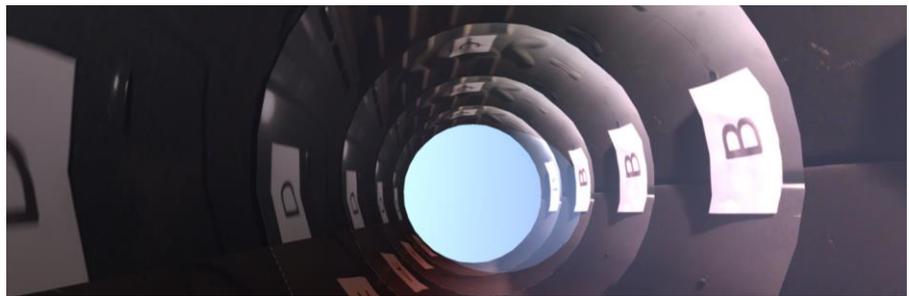
Figura 51. Resultados obtenidos con diferente software de la creación de una imagen panorámica a partir de 4 imágenes. a) Imagen panorámica capturada, b) Resultado obtenido por el script de OpenCV con la función `create_stitcher`, c) Resultados obtenidos por Fiji con el plug-in *Stitching*.

En comparación con la imagen panorámica, al haber logrado capturar imágenes en los 4 ángulos deseados evitando la inclinación sobre el eje Z, los resultados obtenidos tanto por el script de OpenCV como por la aplicación Fiji son similares en un 81%. El porcentaje de similitud se calculó con la librería de Python *scikit-image* que incluye ya la función `compare_ssim` basada en la técnica de *structural similarity measurement (SSM)*<sup>[54]</sup>. Finalmente, se muestra en la Figura 52 el modelo creado a partir de la imagen panorámica previamente construida.

a)



b)



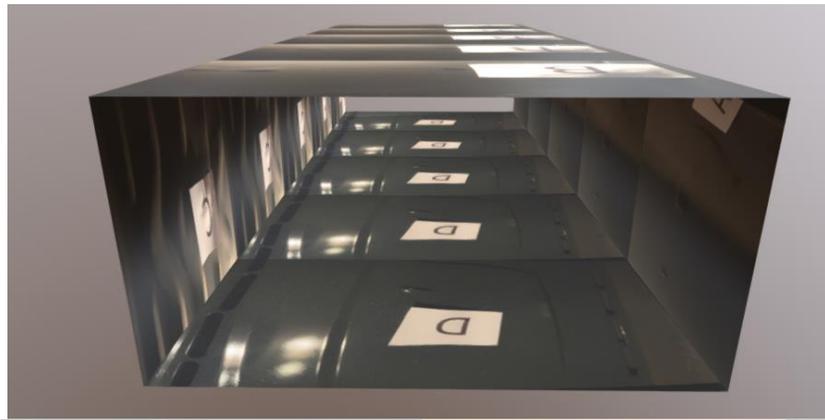
c)



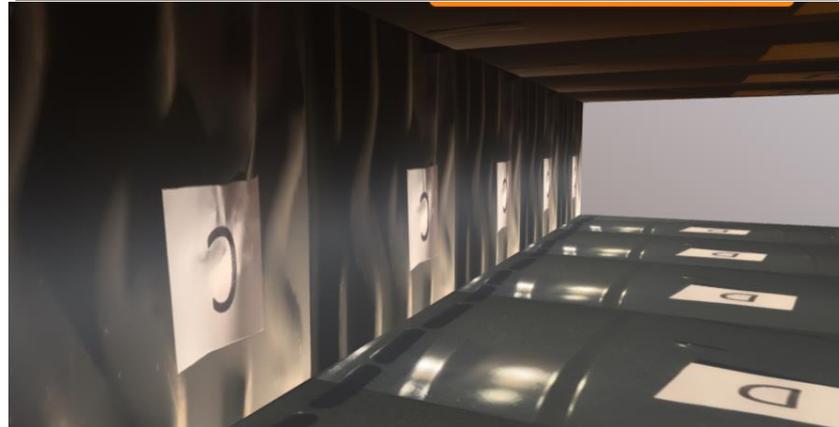
*Figura 52. Visualización del modelo creado a partir de imágenes: a) Vista exterior del modelo, b) Vista desde el interior del modelo, c) Vista de la parte superior del modelo*

A diferencia de la Figura 52, en la Figura 53 observamos la representación haciendo uso de una malla rectangular que se asimila más a la forma del ducto de la cámara de combustión. Sin embargo, para que esta representación se logre adecuadamente es necesario determinar las posiciones adecuadas de las cámaras ya que no todos los lados son iguales. Este correcto posicionamiento puede ser necesario para cada tipo de turbinas por la diferencia en tamaños entre las mismas. La representación cilíndrica es una alternativa más genérica y que además da la información de una manera muy similar a una inspección visual de la actualidad.

a)



b)



c)



Figura 53. Visualización del modelo creado a partir de imágenes: a) Vista exterior del modelo, b) Vista desde el interior del modelo, c) Vista de la parte superior del modelo

### 4.3 PRUEBAS CON IMÁGENES DE LA CÁMARA DE COMBUSTIÓN

No fue posible adquirir imágenes de una cámara de combustión real. Sin embargo, se obtuvieron algunas imágenes a partir de un video de una inspección boroscópica<sup>[51]</sup>. No fue posible obtener imágenes en los diferentes ángulos deseados debido al movimiento y enfoque de la cámara a lo largo del video. Sin embargo, en la Figura 54 se muestran

la agrupación de dos imágenes que contienen información de 3 paredes de la cámara de combustión, localizándose la pared de los inyectores de combustible en la parte central.



a)

b)



c)

*Figura 54. Imágenes de inspección boroscópica de una cámara de combustión. A) Imagen lado izquierdo, B) Imagen lado derecho, C) Stitching de las imágenes.<sup>[3]</sup>*

A través de esta prueba es posible afirmar que el método propuesto es capaz de crear la representación de la cámara de combustión de una turbina aeronáutica. Cabe resaltar que es necesario realizar pruebas adicionales para identificar la mejor configuración de las cámaras en lo referente al posicionamiento de las mismas. En el ejemplo como el del horno, el punto central de cada imagen era la etiqueta, la cual se localizaba justo en el centro de una pared del objeto inspeccionado. Esto significa que las cámaras estaban posicionadas en los ángulos  $0^\circ$ ,  $90^\circ$ ,  $180^\circ$  y  $270^\circ$ . Sin embargo, en el ejemplo mostrado con 2 imágenes de la cámara de combustión, es posible observar que el centro de las imágenes se localiza en la intersección de sus paredes, esto sería equivalente a que las cámaras se posicionaran en los ángulos  $45^\circ$ ,  $135^\circ$ ,  $225^\circ$  y  $315^\circ$ .

respectivamente. Es importante tomar en cuenta que esta última configuración tal vez facilite la identificación de regiones comunes al contar con parte de los componentes característicos de cada pared en las regiones laterales en lugar de en el punto central.

## CONCLUSIONES

Las actividades de inspección boroscópica son actividades visuales de lugares inaccesibles al ojo humano, donde el operador requiere manipular el dispositivo para desplazarlo a lo largo del escenario y posicionarlo de manera adecuada para determinar el estado actual del componente. La solución propuesta es un proceso automatizado de la primera tarea de la actividad de inspección visual que evita la necesidad de contar con un operador especializado para obtener la información interna del componente estudiado.

La solución propuesta hace posible recrear en su totalidad el estado interno del componente inspeccionado realizando la texturización de un modelo con las imágenes capturadas, para dotar al operador de una representación similar a la utilizada actualmente para la determinación del estado del componente. Otras alternativas de conversión a 3D con base en imágenes, a pesar de poder ser representaciones con un mayor nivel de realismo del escenario inspeccionado, agregan complejidad a la tarea de inspección, donde factores como el tiempo son cruciales, y por lo cual no se consideraron, pues requieren un alto tiempo de procesamiento<sup>[29]</sup>.

El método propuesto permitió conseguir la representación en 3D de un escenario capturado. Este método consiste principalmente de tres etapas: adquisición de imágenes, creación de una imagen panorámica y texturización de un modelo. Para la adquisición de imágenes se plantearon dos propuestas para tomar las fotografías y posteriormente procesarlas (en caso de ser necesario) para formar una imagen panorámica. Una vez obtenida la imagen panorámica se crearon de manera automática el conjunto de mallas cilíndricas que forman la representación del escenario de inspección.

El correcto posicionamiento de la cámara y la iluminación del componente inspeccionado fueron desafíos que se abatieron para conseguir obtener resultados aceptables. Las pruebas realizadas en componentes metálicos con algunas características distintivas en sus paredes son una representación similar a la cámara de combustión, lo que hace posible validar la aplicabilidad del método propuesto para la inspección de este componente. Igualmente se lograron realizar pruebas exitosas con

imágenes de una cámara de combustión que ayudaron a validar la aplicabilidad de este trabajo para este tipo de componentes. Sin embargo, es recomendable replicar las pruebas una vez que se cuente con el hardware del dispositivo de inspección; para de esta manera lograr identificar la mejor configuración que asegure la correcta captura de imágenes del componente para su agrupación.

A través del uso de OpenCV y Blender fue posible crear un modelo 3D que representara el estado actual del componente inspeccionado proporcionando al operador la información necesaria para tomar una decisión sobre la necesidad de mantenimiento del componente. Es importante destacar que la solución propuesta es de bajo coste pues no requiere de equipo de cómputo especializado, lo que permite ser ejecutada en cualquier lugar donde se localice la aeronave. A pesar de que la tarea de adquisición de imágenes estaba fuera del alcance de este trabajo, se presentaron propuestas del diseño para su posicionamiento en el robot, para la correcta captura del estado interno del componente a inspeccionar. Se describieron dos configuraciones a considerar de cámaras para la adquisición de imágenes, la primera con una sola cámara sobre un eje rotativo y la segunda con cuatro cámaras sobre un eje fijo. Se recomienda el uso de la configuración con cuatro cámaras pues contiene menos factores a controlar por el robot. El eje rotativo puede ser difícil de controlar lo que complica la correcta adquisición de una imagen panorámica. Es necesario ejecutar algunas pruebas preliminares una vez que se tenga el dispositivo completo. De igual forma los algoritmos presentados pueden ser utilizados para pruebas similares en otros sectores industriales tomando en cuenta las consideraciones descritas en este trabajo como la alineación de las imágenes.

Con base en lo descrito por este trabajo se puede aceptar la hipótesis planteada ya que es posible aplicar el método propuesto en pruebas de inspección aeronáuticas, atacando la principal restricción de acceso limitado, razón por la cual estas pruebas suelen requerir el desmontaje de la turbina del ala de la aeronave. La representación 3D puede ser vista como un modelo digital utilizado para pruebas no destructivas que otorgue a un experto una visualización con la información necesaria para detectar desperfectos en un componente sin la necesidad de ejecutar la inspección. A través de esta representación se evita la necesidad de que un experto realice la inspección y se enfoque solamente a la interpretación de los resultados. Esto ayuda a la reducción de

tiempos operacionales relacionadas con actividades de desplazamiento tanto del experto como de la turbina, lo cual eficientiza el nivel operativo de la turbina al evitar remociones necesarias en caso de no requerir ningún tipo de mantenimiento.

## **APORTACIÓN DE LA TESIS**

El uso del software propuesto en este trabajo junto con un robot que pueda acceder fácilmente a ciertos componentes aeronáuticos de acceso restringido en ala elimina la necesidad de realizar actividades operativas que adicionan tiempos innecesarios a las tareas de inspección visual. De esta manera se eficientiza el uso operación del motor al crear una alternativa de confirmación de falla en el componente inspeccionado sin necesidad de remoción del motor. El modelo 3D crea una representación equivalente a la obtenida durante una inspección boroscópica convencional, permitiendo al especialista del componente determinar la necesidad de mantenimiento aun no habiendo realizado la inspección.

## RECOMENDACIONES

La presente investigación presenta diferentes ramificaciones para investigaciones futuras enfocadas a puntos que aún no se encuentran definidos en su totalidad sobre todo con respecto al robot inspector.

En este trabajo se describieron posibles configuraciones de las cámaras para la tarea de adquisición de datos. Una línea de investigación es con respecto al mecanismo utilizado para el posicionamiento de las cámaras con cada una de estas configuraciones. Mientras la configuración con cuatro cámaras requiere de un mecanismo fijo, es importante estudiar el mejor ángulo de cada cámara para lograr la creación de una imagen panorámica adecuada. Por otro lado, en el caso de la configuración con una sola cámara es importante determinar las conexiones necesarias para girar el eje donde se ubique la cámara, así como la velocidad de rotación para conseguir obtener una imagen panorámica que pueda ser utilizada para la texturización del modelo.

Es importante también analizar posibles escenarios de falla del sistema e identificar como atacarlos para asegurar la adquisición de toda la información requerida. Un ejemplo sería la descompostura de una cámara. Esto podría atacarse teniendo una segunda cámara en la misma posición que pudiera utilizar en caso de fallar la primera cámara. Para esto se requiere monitorear la validez de cada cámara para determinar si son aptas para su uso durante la inspección y en caso de no serlo comenzar a capturar las imágenes con su cámara equivalente. Todo esto le dará redundancia al sistema haciéndolo más confiable.

Otras líneas de investigación son el módulo de desplazamiento y la iluminación del espacio inspeccionado. El módulo de desplazamiento no solo se basa en el envío de las señales correspondientes para hacer que el robot recorra el ducto donde se encuentre, sino también está relacionado con la velocidad de captura de imágenes para asegurar adquirir toda la información del escenario. En lo que respecta a la iluminación se mencionaron las opciones de un anillo de iluminación compartido por todas las cámaras y la posibilidad de que cada cámara tenga su propio flash. En el caso del anillo es importante determinar si se debe mantener encendido a lo largo de toda la prueba de inspección o si es mejor que parpadee cada determinado tiempo. Todo esto es con el

objetivo de que la luz se vuelva un factor positivo que nos permita obtener imágenes claras que resulten fáciles de agrupar.

Recordando que este trabajo se enfocó en la primera parte de la actividad de inspección visual, teniendo como uno de sus objetivos automatizar la tarea de captura y representación del escenario para su futuro análisis, es importante realizar un estudio que permita que esta actividad sea realizada de manera automática completamente. Esto significa añadir la funcionalidad de detección de fisuras o grietas en las paredes de estos componentes para prescindir de la necesidad de un operador en cualquiera de las etapas de inspección. Actualmente existen soluciones con enfoques similares que se podrían utilizar de base para complementar este trabajo. Sin embargo, es importante considerar que este nivel de solución requiere altos casos de prueba y pasar por un proceso de certificación de las entidades para siempre poder asegurar que el componente inspeccionado sea capaz de continuar operando.

Es posible mejorar la presentación del modelo creado a través de un motor de tercera dimensión. En artículos consultados para esta investigación se menciona el motor de Unity como el de mayor trascendencia. Además, es importante analizar los tiempos de integración del modelo a este motor para evitar incrementar los tiempos totales de la inspección. Todo esto podría integrarse en un proyecto de realidad virtual que pudiera sustituir la forma de realizar las inspecciones visuales hoy en día.

Finalmente, parte de los objetivos era crear una solución que pudiera ser aplicada en otros sectores industriales. A través del uso de vehículo autónomos se podría integrar la solución propuesta para actividades de inspección en áreas industriales o de ingeniería civil. En el caso de las áreas industriales, podría ser utilizado para el análisis de partes de una fábrica que son difíciles de acceder por una persona, no necesariamente por el tamaño, sino por la altura. Un ejemplo de esto son las chimeneas de una fábrica. Para el sector de ingeniería civil, el uso de esta solución en conjunto con vehículos autónomos podría ayudar al análisis de estructuras arquitectónicas.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Katunin, A., Dragan, K., Dziendzikowski, M. Damage identification in aircraft composite structures: A case study using various non-destructive testing techniques. *Composite structures*, 2015, vol. 127, p. 1-9.
- [2] Gholizadeh, S. A review of non-destructive testing methods of composite materials. *Procedia structural integrity*, 2016, vol. 1, p. 50-57.
- [3] Usamentiaga, R., Venegas, P., Guerediaga, J., Molleda, J., Bulnes, F. G. Infrared thermography for temperature measurement and non-destructive testing. *Sensors*, 2014, vol. 14, no 7, p. 12305-12348.
- [4] IATA. Industry Statistics.[en línea][Consultado 1 enero 2019] Disponible en: <https://www.iata.org/publications/economics/Reports/Industry-Economic-Performance/Airline-Industry-Economic-Performance-December-18-Datatables.pdf>.
- [5] Kumar, U., Crocker, J., Knezevic, J. El-Haram, M. Reliability, maintenance and logistic support: a life cycle approach. Dordrecht: Kluwer Academic Publisher Group, 2000.
- [6] Rodrigues, L. R., Yoneyama, T., Junior, C. L. N. How aircraft operators can benefit from PHM techniques. 2012 IEEE Aerospace Conference. 2012. DOI 10.1109/aero.2012.6187376.
- [7] Panigrahi, S. K., Sarangi, N. Aero Engine Combustor Casing: Experimental Design and Fatigue Studies. CRC Press, 2017.
- [8] Vembar, D., Sadasivan, S., Duchowski, A. T., Stringfellow, P., Gramopadhye, A. Design of a virtual borescope: a presence study. In *Proceedings of HCI International*, pp. 22-27. 2005.
- [9] Colin, D. Human factors, good practices in borescope inspection. 2018.
- [10] Pereira, C., Pereira, F. Embedded Image Processing Systems for Automatic Recognition of Cracks using UAVs. *IFAC 2015*.
- [11] Rani, U., Nitasha, N. Image Segmentation using Mathematical Morphology: A Study. *International Journal of Advanced Research in Computer Science*, 11, 2016, vol. 7, no. 6 ProQuest Central.
- [12] Haralick, R. Image Analysis Using Mathematical Morphology. *IEEE Transactions on pattern analysis and machine intelligence*. 4 July 1987.

- [13] Zhang, W., Zhang, Z., Qi, D., and Liu, Y. Automatic Crack Detection and Classification Method for Subway Tunnel Safety Monitoring. *Sensors*, 2014, vol. 14, no. 10. pp. 19307-19328 Medical.
- [14] Maini, R., and Himanshu, A. Study and comparison of various image edge detection techniques. *International journal of image processing (IJIP)* 3, no. 1 (2009): 1-11.
- [15] Shrivakshan, G. T., Chandrasekar, C. A comparison of various edge detection techniques used in image processing. *International Journal of Computer Science Issues (IJCSI)* 9, no. 5 (2012): 269.
- [16] Baradeswaran, A., Selvakumar, J., Padma, P. Reconstruction of Images into 3D Models using CAD Techniques. *European Journal of Applied Engineering and Scientific Reserach*. 2014.
- [17] Moons, T., Van Gool, L., Vergauwen, M. 3D Reconstruction from Multiple Images. *Foundation and Trends in Computer Graphics and Vision* 2009.
- [18] Yang, B., Huang, R., Li, J., Tian, M., Dai W., Zhong R. Automated Reconstruction of Building LoDs from Airborne LiDAR Point Clouds Using an Improved Morphological Scale Space. *Remote Sensing* 2017;9(1):14.
- [19] Elbehiery, H., Hefnawy, A., Elewa, M. Surface defects detection for ceramic tiles using image processing and morphological techniques. (2005).
- [20] Fryer, T. It's a bug's life as a maintenance engineer. *Engineering & Technology*, 2018, vol. 13, no 11/12, p. 46-47.
- [21] Rus, D., Tolley, M. T. Design, fabrication and control of soft robots. *Nature*, 2015, vol. 521, no 7553, p. 467.
- [22] Roemer, M. J., Nwadiogbu, E. O., Bloor, G. Development of diagnostic and prognostic technologies for aerospace health management applications. En 2001 IEEE Aerospace Conference Proceedings (Cat. No. 01TH8542). IEEE, 2001. p. 3139-3147.
- [23] Pereira, E. Aircraft Engines Maintenance Cost and Reliability. An Appraisal of the Decision Process to Remoe an Engine for a Shop Visit Aiming at Minimum Maintenance Unit Cost. Universidade Nova de Lisboa. 2012.

- [24] Mix, P. E. Introduction to Nondestructive Testing A Training Guide. Wiley-Interscience. 2005.
- [25] Sobrino, J. A. S. Introducción a la fotogrametría. Madrid: ETSI, 2006.
- [26] Caro, J. L. Fotogrametría y modelado 3D: un caso práctico para la difusión del patrimonio y su promoción turística. 2012.
- [27] Beraldin, J. A., Picard, M., El-Hakim, S., Bandiera, A. Virtualizing a Byzantine crypt by combining high-resolution textures with laser scanner 3D data. 2002.
- [28] Wang, Weixing, et al. Applications of terrestrial laser scanning for tunnels: a review. Journal of Traffic and Transportation Engineering (English Edition), 2014, vol. 1, no 5, p. 325-337.
- [29] Falkingham, Peter L. Acquisition of high-resolution three-dimensional models using free, open-source, photogrammetric software. Palaeontologia electronica, 2011, vol. 15, no 1, p. 1-15.
- [30] Thirion, J. P. The Extremal mesh and the understanding of 3D surfaces. [Research Report] RR-2149, INRIA. 1993, pp.30. ffinria-00077150f.
- [31] Ding, K., Ye, L. Simulation of laser shock peening on a curved surface. In Woodhead Publishing Series in Metals and Surface Engineering, Laser Shock Peening, Woodhead Publishing, 2006, Pages 133-150, ISBN 9781855739291.
- [32] Locker, A. Best 3D Design/3D Modeling Software 2019 (15 of 30 are Free) | All3DP, 2019. All3DP[en línea].[Consultado 12 febrero 2019]. Disponible en: <https://all3dp.com/es/1/mejores-programas-diseno-3d-software-modelado-3dgratis/>.
- [33] CFM International. Non-Destructive Test Manual Part 7: Borescope Inspection. May 31, 1999.
- [34] Stewart, F., Cox, B., Vorstius, J., Verbeni, A., Qiu, Y., Cochran, S. Capsule-based ultrasound-mediated targeted gastrointestinal drug delivery. En 2015 IEEE International Ultrasonics Symposium (IUS). IEEE, 2015. p. 1-4.
- [35] Conlan C. (2017) The Blender Interface. In: The Blender Python API. Apress, Berkeley, CA.
- [36] Donato, J. M. T., Famador, S. M. W. Classifying Human and Animal Hair Using Probabilistic Neural Networks for Texture Classification. 2017.

- [37] Preibisch, S., Saalfeld, S., Tomancak, P. Globally optimal stitching of tiled 3D microscopic image acquisitions. *Bioinformatics*, 2009, vol. 25, no. 11, p. 1463-1465.
- [38] Wald, M. L. FAA Official Scrapped Tape of 9/11 Controllers' Statements. *New York Times*, 2004.
- [39] Legland, D., Arganda-Carreras, I., Andrey, P. MorphoLibJ: integrated library and plugins for mathematical morphology with ImageJ. *Bioinformatics*, 2016, vol. 32, no. 22, p. 3532-3534.
- [40] Kovesi, P. D. MATLAB and Octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia, available from: <http://www.csse.uwa.edu.au/~pk/research/matlabfns>, 2000, vol. 147, p. 230.
- [41] Beck, T., Plessier, J. Y., Pagano, I., Wildi, F. The CHEOPS (characterising exoplanet satellite) mission: telescope optical design, development status and main technical and programmatic challenges. En *International Conference on Space Optics—ICSO 2016*. International Society for Optics and Photonics, 2017. p. 1056218.
- [42] Garnier, C., Pastor, M. L., Eyma, F., Lorrain, B. The detection of aeronautical defects in situ on composite structures using Non-Destructive Testing. *Composite structures*, 2011, vol. 93, no 5, p. 1328-1336.
- [43] Addepalli, Sri, Roy, R., Axinte, D., Mehnen, J. 'In-situ' inspection technologies: Trends in degradation assessment and associated technologies. *Procedia CIRP*, 2017, vol. 59, p. 35-40.
- [44] Dong, X., Axinte, D., Palmer, D., Cobo, S., Raffles, M., Rabani, A., Kell, J. Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines. *Robotics and Computer-Integrated Manufacturing*, 2017, vol. 44, p. 218-229.
- [45] Stent, S. A. I., et al. A low-cost robotic system for the efficient visual inspection of tunnels. En *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*. IAARC Publications, 2015. p. 1.
- [46] Zhao, J.-C.; WESTBROOK, J. H. Ultrahigh-temperature materials for jet engines. *MRS bulletin*, 2003, vol. 28, no 9, p. 622-630.
- [47] Sattar, T., et al. Mobile wall climbing and swimming robots to inspect aircraft, storage tanks, pressure vessels and large infrastructure. 2008.

- [48] Testo 882 Infrared Camera [en-línea] [Consultado 12 febrero 2018]. 2018. Malasya : Testo. Disponible en: <https://www.testo.com/en-MY/testo-882/p/0560-0882>.
- [49] Ruiz, J. Mejoras a la Inspección Boroscópica del Motor Turbofan CFM56-7B. 2013. Tesis de Licenciatura. Universitat Politècnica de Catalunya.
- [50]
- [51] Micro Scout Cam [en línea]. 2019. Omer, Israel : ScoutCam Ltd. [Consultado 13 January 2019]. Disponible en: <https://www.scoutcam.com/products-and-technology/1-2mm-disposable-microcamera/>.
- [52] GE Oil & Gas Digital Solutions. "Aircraft Engine Inspection with GE's Video Borescope" [Consultado: 9 de septiembre 2019] Youtube. Disponible en: <https://www.youtube.com/watch?v=VATyEPvuQy8>.
- [53] Rus, D.; Tolley, M. Design, fabrication and control of soft robots. *Nature*, 2015, vol. 521, no 7553, p. 467.
- [54] Schindelin, J., et al. Fiji: an open-source platform for biological-image analysis. *Nature methods*, 2012, vol. 9, no 7, p. 676.
- [55] Wang, Z. et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 2004, vol. 13, no 4, p. 600-612.

## ANEXOS

### ANEXO A

#### Código en Python para la representación 3D en Blender con malla cilíndrica.

Este trabajo utilizó una malla primitiva de forma cilíndrica como base para la representación del interior de una cámara de combustión. Para conseguir esto, se desarrolló el código mostrado a continuación a través del cual se logra la creación de múltiples mallas que en su conjunto representan el escenario inspeccionado.

```
1 import bpy
2 import bmesh
3 import os
4
5 #bpy.ops.object.mode_set(mode='OBJECT')#set object mode
6 bpy.ops.object.delete(use_global=False)
7 def ls(cdir): # return directory contents of cdir
8     ls = os.listdir(cdir)
9     return ls
10
11sc = bpy.data.scenes[0] # get current scene
12imgdir="C:\\Users\\Maduch\\Documents\\Maestria_SIM\\Tesis\\Imagenes\\Test\\
\\Test5"
13img_ext='jpg'
14
15files=[] # list for files
16images=[] # list for image files
17files=(ls(imgdir)) # read file list into list
18file_count=len(files) # file count
19
20for a in range(0,file_count): # for each file:
21     if files[a].endswith(img_ext): # does it end with?
22         images.append(files[a]) # if so then add to images list
23
24bpy.ops.file.autopack_toggle()
25image_count=len(images) # count of images
26print(file_count)
27print(image_count)
28pos = -6 #initial position of first mesh (needs to be
adjusted for real application)
29for a in range(0,image_count): # for each image
30     print("=====")
31     print('loop count: '+str(a))
32     # CREATE MESH
33     # end_fill_type = NOTHING to make a hollow cylinder
34     # radius corresponds to define the radius size of the cylinder
35     # location(x,y,z) location on y to be increased for each new panorama
36     # rotation 90 degrees in order to make a tunnel form
37     bpy.ops.mesh.primitive_cylinder_add(end_fill_type='NOTHING',
radius=2,location=(0,pos,0), rotation=(1.57,0,0))
38
39     bpy.ops.object.mode_set(mode='OBJECT', toggle=False) #set context to
object mode
40     bpy.ops.object.shade_smooth() #smooth property displays the mesh
without showing the edges
```

```

41 plane=bpy.context.scene.objects.active
42 bpy.ops.object.mode_set(mode='EDIT')#set edit mode to add texturization
43
44 bpy.ops.mesh.flip_normals()          #flip normal to fill mesh inside
45
46 # force edge selection mode
47 bpy.context.tool_settings.mesh_select_mode = (False, True, False)
48
49 # this will select every 'even' numbered edge from the list of edges.
50 obj = bpy.context.active_object
51
52 bm = bmesh.from_edit_mesh(obj.data)
53 for edge in bm.edges:
54     if edge.index == 66:
55         edge.select = True
56         continue
57
58     edge.select = False
59
60 # Definition of how the mesh is going to be unwrapped
61 bpy.ops.mesh.mark_seam(False)
62 bpy.ops.mesh.select_all(action='SELECT')
63 bpy.ops.uv.unwrap()
64
65 imgpath=imgdir+'\\'+images[a]          # make string with path ti image
66 img = bpy.data.images.load(imgpath)    # load image
67 bpy.ops.image.open(filepath=imgpath, directory=imgdir)    # open image
68 new_image = bpy.data.images[images[a]]
69 bpy.data.screens['UV Editing'].areas[1].spaces[0].image = new_image
#put image in UV Editor
70 last_area = bpy.context.area.type
71 bpy.context.area.type = 'IMAGE_EDITOR'
72
73 # perform transformation fot better display results
74 bpy.ops.transform.translate(value=(0.0, 0.4, 0.0))    #0.14
75 bpy.ops.transform.resize(value=(1.0, img.size[1]/22.5, 1.0))    #/1300
76
77 bpy.context.area.type = last_area
78
79 mat=bpy.data.materials.new('mat'+str(a))          #Define Material name
for texture
80 bpy.context.object.data.materials.append(mat)
81
82 tex=bpy.data.textures.new('ColorTex', type = 'IMAGE')
83 imgpath=imgdir+'\\'+images[a]          # make string with path ti image
84 img = bpy.data.images.load(imgpath)    # load image
85 tex.image = img
86 tex.extension = 'CLIP'
87 mtex = mat.texture_slots.add()
88 mtex.texture = tex          # add texture
89 mtex.texture_coords = 'UV'    #change to OBJECT and select Cylinder
90
91 mtex.mapping= 'FLAT'
92 imgX = img.size[0]/1000.0          # calculate dimensions
93 imgY = img.size[1]/1000.0
94
95 pos = pos + 2
96 bpy.ops.object.mode_set(mode='OBJECT') #set object mode
97

```

## ANEXO B

### Código en Python para la representación 3D en Blender con malla rectangular.

El ducto por el cual se realiza la inspección visual de una cámara de combustión presenta una forma rectangular. El código descrito en este anexo fue desarrollado para la creación de un modelo con una malla primitiva de tipo rectangular que pudiera representar este componente.

```
1: import bpy
2: import bmesh
3: import os
4:
5: #bpy.ops.object.mode_set(mode='OBJECT')           #set object mode
6: bpy.ops.object.delete(use_global=False)
7: def ls(cdir):                                     # return directory contents
of cdir
8:     ls = os.listdir(cdir)
9:     return ls
10:
11: sc = bpy.data.scenes[0]                           # get current scene
12:
imgdir="C:\\Users\\Maduch\\Documents\\Maestria_SIM\\Tesis\\Imagenes\\Test\\T
est6"
13: img_ext='png'
14:
15: files=[]                                         # list for files
16: images=[]                                       # list for image files
17: files=ls(imgdir)                                # read file list into list
18: file_count=len(files)                           # file count
19:
20: for a in range(0,file_count):                    # for each file:
21:     if files[a].endswith(img_ext):               # does it end with?
22:         images.append(files[a])                 # if so then add to images
list
23:
24: bpy.ops.file.autopack_toggle()
25: image_count=len(images)                          # count of images
26: print(file_count)
27: print(image_count)
28: pos = -6                                         #initial
position of first mesh (needs to
29:                                                 #be
adjusted for real application)
30: for a in range(0,image_count):                   # for each image
31:     print("=====")
32:     print('loop count: '+str(a))
33:     # CREATE MESH
34:     # end_fill_type = NOTHING to make a hollow cylinder
35:     # radius corresponds to define the radius size of the cylinder
36:     # location(x,y,z) location on y to be increased for each new panorama
37:     # rotation 90 degrees in order to make a tunnel form
38:     bpy.ops.mesh.primitive_cube_add(location=(0,pos,0))
39:
```

```

40:     bpy.ops.object.mode_set(mode='OBJECT', toggle=False) #set context to
object mode
41:     bpy.ops.object.shade_smooth()           #smooth property displays the mesh
without showing the edges
42:     plane=bpy.context.scene.objects.active
43:     bpy.ops.object.mode_set(mode='EDIT')           #set edit mode
to perform texturization
44:
45:     bpy.ops.mesh.flip_normals()             #flip normal to fill mesh inside
46:
47:     # force face selection mode
48:     bpy.context.tool_settings.mesh_select_mode = (False, False, True)
49:
50:     # this will select every 'even' numbered edge from the list of edges.
51:     obj = bpy.context.active_object
52:
53:     bm = bmesh.from_edit_mesh(obj.data)
54:     bpy.context.active_object.dimensions[0] = 4
55:     bm.faces.ensure_lookup_table()
56:     #delete front and back faces of the cube
57:     for face in bm.faces:
58:         face.select = False
59:     bm.faces[1].select = True
60:     bm.faces[3].select = True
61:     bpy.ops.mesh.delete(type='ONLY_FACE')
62:
63:     # force edge selection mode
64:     bpy.context.tool_settings.mesh_select_mode = (False, True, False)
65:     for edge in bm.edges:
66:         if edge.index == 66:
67:             edge.select = True
68:             continue
69:
70:         edge.select = False
71:
72:     # select FACE mode
73:     bpy.context.tool_settings.mesh_select_mode = (False, False, True)
74:
75:     bpy.context.area.type = 'IMAGE_EDITOR'
76:     # Definition of how the mesh is going to be unwrapped
77:     bpy.ops.mesh.select_all(action='SELECT')
78:
79:     #bpy.ops.uv.sort_elements(type='VIEW_ZAXIS', elements={'FACE'})
80:     bpy.ops.mesh.mark_seam(False)
81:     bpy.ops.uv.unwrap()
82:     bpy.ops.uv.unwrap()
83:
84:     #switch rectangle faces for correct image texturization with panorama
85:     bm.faces.ensure_lookup_table()
86:     for face in bm.faces:
87:         face.select = False
88:     bm.faces[0].select = True
89:     bm.faces[3].select = True
90:
91:     bpy.ops.transform.mirror(constraint_axis=(True, False, False))
92:
93:     bpy.ops.mesh.select_all(action='SELECT')
94:
95:

```

```

96:   imgpath=imgdir+'\\'+images[a]           # make string with
path ti image
97:   img = bpy.data.images.load(imgpath)     # load image
98:   bpy.ops.image.open(filepath=imgpath, directory=imgdir) # open
image
99:   new_image = bpy.data.images[images[a]]
100:  bpy.data.screens['UV Editing'].areas[1].spaces[0].image = new_image
#put image in UV Editor
101:  last_area = bpy.context.area.type
102:  bpy.context.area.type = 'IMAGE_EDITOR'
103:
104:  # perform transformation fot better display results
105:  bpy.ops.transform.translate(value=(0.90, 0.0, 0.0)) #0.14
106:  bpy.ops.transform.resize(value=(1.45, img.size[1]/630, 1.0)) #/1300
107:
108:  bpy.context.area.type = last_area
109:
110:  mat=bpy.data.materials.new('mat'+str(a)) #Define Material name
for texture
111:  bpy.context.object.data.materials.append(mat)
112:
113:  tex=bpy.data.textures.new('ColorTex', type = 'IMAGE')
114:  imgpath=imgdir+'\\'+images[a]           # make string with
path ti image
115:  img = bpy.data.images.load(imgpath)     # load image
116:  tex.image = img
117:  tex.extension = 'CLIP'
118:  mtex = mat.texture_slots.add()
119:  mtex.texture = tex # add texture
120:  mtex.texture_coords = 'UV' #change to OBJECT and select Cylinder
121:
122:  mtex.mapping= 'FLAT'
123:  imgX = img.size[0]/1000.0 # calculate dimensions
124:  imgY = img.size[1]/1000.0
125:
126:  pos = pos + 2
127:  bpy.ops.object.mode_set(mode='OBJECT') #set object
mode
128:
129: #bpy.ops.export.sketchfab()
130:
131:
132:

```