



UNIVERSIDAD
DE ATACAMA

FACULTAD DE INGENIERÍA
DEPTO. DE ING. INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN

**INSPECCIÓN ACTIVA DE OBJETOS PARA
LA DETECCIÓN Y CUANTIFICACIÓN DEL
VOLUMEN DE UN EVENTUAL DAÑO,
USANDO RECONSTRUCCIÓN 3D**

Trabajo de titulación presentado como parte de los requisitos para optar al título
de Ingeniero Civil en Computación e Informática

Profesor Guía: Vladimir Riffo Bouffanais

Rodrigo Sebastián Hidalgo Avila

Copiapó, Noviembre 2023, Chile.



UNIVERSIDAD
DE ATACAMA

FACULTAD DE INGENIERÍA
DEPTO. DE ING. INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN

**INSPECCIÓN ACTIVA DE OBJETOS PARA
LA DETECCIÓN Y CUANTIFICACIÓN DEL
VOLUMEN DE UN EVENTUAL DAÑO,
USANDO RECONSTRUCCIÓN 3D**

Trabajo de titulación presentado como parte de los requisitos para optar al título
de Ingeniero Civil en Computación e Informática

Profesor Guía:
Vladimir Riffo Bouffanais

Miembros del Comité:
Héctor Cornide Reyes
John Castro Llanos

Rodrigo Sebastián Hidalgo Avila

Copiapó, Noviembre 2023, Chile.

*Este trabajo de titulación se
lo dedico a mi familia y en
especial a mi madre por
todos sus consejos y
enseñanzas. Gracias a ello
he logrado avanzar y llegar
hasta aquí.*

Agradecimientos

En primer lugar quiero agradecer a mi profesor guía Vladimir Riffo B., ya que su enseñanza y apoyo fueron indispensables durante cada una de las etapas que conforman la elaboración de este trabajo de titulación y así poder conseguir todos los resultados esperados.

También agradezco al Departamento de Ingeniería Informática y Ciencias de la Computación de la Universidad de Atacama por permitirme utilizar parte de sus instalaciones. Dicho aporte fue de suma importancia para avanzar en mi estudio universitario.

Por último, agradezco a mi familia y amigos por toda la dedicación que tuvieron conmigo, dándome consejos e inculcándome valores, los cuáles me han ayudado a ser una persona de bien.

Resumen

Los procesos de inspección automatizados ofrecen ventajas de objetividad y eficiencia para cada prueba. Por este motivo, se han desarrollado determinadas técnicas de procesamiento digital de imágenes y visión por computador. Sin embargo, sigue siendo una pregunta abierta debido a la gran variabilidad en apariencia y forma de los objetos de estudio. En este trabajo, proponemos una metodología de inspección que nos permite analizar objetos. Para lo cual hemos desarrollado un sistema de inspección activa para detectar daños o anomalías. Su funcionamiento comienza con la adquisición de imágenes del objeto mediante dos cámaras ubicadas frente a una plataforma giratoria, con rotación controlada, y proyectando una línea láser sobre el objeto. A continuación se muestra la reconstrucción 3D de una sección del objeto, para luego realizar una comparación, mediante un descriptor 3D y análisis de distancia entre puntos del modelo 3D generado del objeto de prueba y un modelo 3D de una versión del mismo objeto sin daño, el cual fue almacenado en una base de datos. Si se detecta algún daño, se continúa con la inspección hasta que se completa toda la sección dañada, y finalmente se calcula su volumen. Las principales contribuciones de nuestro trabajo se resumen en: *i*) Combinación eficiente de algoritmos de visión por computador para la estimación de volumen, *ii*) Inspección activa en la búsqueda automática de daño superficial en objetos, y *iii*) Medición del volumen de daño detectado en unidades métricas, que puede ser un daño sustractivo o aditivo. Para validar nuestra propuesta, hemos aplicado esta metodología a diferentes objetos, pudiendo así estimar el volumen de daño (si lo hay). Nuestros resultados son prometedores, ya que logramos un error porcentual absoluto medio, $MAPE = 2,78\%$.

Palabras Clave: Cálculo de volumen, Inspección activa, Detección de daño, Reconstrucción 3D, Inspección no destructiva.

Abstract

Automated inspection processes offer the advantages of objectivity and efficiency for each test. For this reason, certain techniques for digital image processing and machine vision have been developed. However, it remains an open question due to the great variability in appearance and shape of the objects of study. In this work, we propose an inspection methodology that allows us to analyze objects. For which we have developed an active inspection system to detect damage or malfunction. Its operation begins with the acquisition of images of the object using two cameras located in front of a rotating platform, with controlled rotation, and projecting a laser line on the object. The following is the 3D reconstruction of a sector of the object, to then make a comparison, using a 3D descriptor and distance analysis between points of the 3D test model and the 3D model generated with a base version without damage, of the same object, stored in a database. If any damage is detected, the inspection is continued until the entire damaged sector is completed, and finally its volume is calculated. The main contributions of our work are summarized as: *i*) Efficient combination of machine vision algorithms for volume estimation, *ii*) Active inspection for the automatic search of surface damage on objects, and *iii*) Measurement of the volume of damage detected in metric units, which can be subtractive or additive damage. To validate our proposal, we have applied this methodology to different objects, and thus be able to estimate the volume of damage (if any). Our results are promising, as we achieved a root mean squared error, $MAPE = 2,78\%$.

Keywords: Volume measurement, Active inspection, Damage detection, 3D Reconstruction, Non-Destructive testing.

Índice general

Índice general	v
Índice de figuras	vii
Índice de tablas	viii
1. Introducción	1
1.1. Hipótesis	3
1.2. Objetivos	3
1.2.1. Objetivo General	3
1.2.2. Objetivos Específicos	3
1.3. Contribución	4
2. Trabajos Relacionados	5
3. Método Propuesto	10
3.1. Descripción de la metodología	10
3.2. Método de Inspección	14
3.3. Adquisición de Imágenes	14
3.4. Pre-procesamiento de Imágenes	16
3.5. Reconstrucción 3D	20
3.6. Alineación de Nubes de Puntos 3D	23
3.7. Detección de Daño	24
3.8. Cálculo de Volumen	27
4. Evaluación	30
4.1. Diseño del Sistema	30
4.2. Pruebas de Desempeño	31

4.2.1. Inspección a objetos sin daño	31
4.2.2. Inspección a objetos con daño aditivo	32
4.2.3. Inspección a objetos con daño sustractivo	33
4.3. Análisis de Resultados	34
5. Conclusiones	38
Referencias	40

Índice de figuras

3.1.	Sistema de inspección para la detección de un eventual daño.	15
3.2.	Segmentación de la línea láser. (a) Canal rojo de <i>ND_LOn</i> , (b) Canal rojo de <i>ND_LOff</i> , (c) Resta de canales de rojo (<i>R_LOn</i> – <i>R_LOff</i>).	17
3.3.	Esqueletización de la línea segmentada. (a) Binarización de <i>Img_Laser</i> , (b) Erosión utilizando un disco de 1 mm de radio, y (c) Esqueletización.	18
3.4.	Alineación de Objetos. (a) Objeto base, <i>SObj</i> , (b) Objeto de prueba, <i>TObj</i> , (c) Objeto base y de prueba alineados.	24
3.5.	Representación de <i>AlphaShape</i> . (a) Región del objeto Pato reconstruida mediante nube de puntos, (b) Región del objeto Pato creada utilizando <i>AlphaShape</i> sobre la nube de puntos.	28
4.1.	Sistema detector de daño.	30
4.2.	Objetos de prueba. (a) Cono, (b) Pato, (c) León, (d) Conejo Stanford, y (e) Tetera.	32
4.3.	Cubos de plastilina de 1000 [mm ³], 3375 [mm ³] y 8000 [mm ³].	32
4.4.	Objetos con daño aditivo. (a) Conejo Stanford AD1, (b) Tetera AD2, y (c) Cono AD3.	33
4.5.	Objetos con daño sustractivo. (a) Pato SD1, (b) Conejo Stanford SD2, y (c) Cono SD3.	33
4.6.	Reconstrucción 3D de daño aditivo para las pruebas: (a) Conejo Stanford AD1, (b) Tetera AD2, y (c) Cono AD3.	34
4.7.	Reconstrucción 3D de daño sustractivo para las pruebas: (a) Duck SD1, (b) Stanford Bunny SD2, y (c) Cone SD3.	35

Índice de tablas

4.1. Pruebas de rendimiento para el sistema de detección de daño.	36
---	----

Capítulo 1

Introducción

“Eureka”, habría sido la palabra usada por el matemático griego *Arquímedes de Siracusa* tras descubrir el que ahora se denomina *Principio de Arquímedes*, que establece la relación entre el volumen de un cuerpo sumergido y la fuerza de flotación que éste experimenta. Lo anterior demuestra la importancia y complejidad de determinar el volumen de un objeto y así comprobar su integridad.

La inspección visual de objetos se está utilizando cada vez más en los sectores manufactureros más diversos, aumentando los factores de calidad en los productos elaborados. Sin embargo, esta tarea aún la llevan a cabo inspectores humanos y con un apoyo tecnológico limitado.

La inspección de objetos como tarea humana es vulnerable a errores ya que puede ser exigente, estresante, aburrida, difícil e incierta. Debido a ello, la automatización de los sistemas de inspección se vuelve cada vez más indispensable para mejorar la efectividad de los resultados [See *et al.*, 2017]. La literatura muestra que en el área de la inspección visual automática de objetos se han generado diferentes enfoques, dependiendo de la aplicación [Czimmermann *et al.*, 2020]. Sin embargo, aún hay problemas que deben abordarse: *i) pérdida de generalidad*, esto se debe a que los enfoques desarrollados para una aplicación no pueden ser usados para otras aplicaciones; *ii) precisión de detección deficiente*, lo que significa la existencia de una compensación fundamental entre los falsos positivos (falsas alarmas) y la pérdida de detecciones; *iii) limitada robustez*, dado que los requisitos para el uso de un método a menudo se cumplen solo para estructuras simples; y *iv) baja adaptabilidad*, ya que puede ser muy difícil modificar el diseño de un sistema automático [Riffo y Mery, 2016].

En los últimos años han mejorado los métodos visuales de inspección y detección de daño¹. mediante sistemas que se enfocan en el uso de redes neuronales artificiales [Nikonova *et al.*, 2021; Vundekode *et al.*, 2021] o aquellos que utilizan la visión estereoscópica y el escaneo con proyección de línea láser, junto con diseños que permiten una mayor movilidad del objeto a ser analizado, para así lograr una reconstrucción 3D más completa [Isa y Lazoglu, 2017; Liu *et al.*, 2020; Wang *et al.*, 2020].

Sin embargo, estos últimos tipos de sistemas generalmente tienen algunas desventajas como operación errática, una gran cantidad de tiempo en su ejecución, uso limitado y la necesidad de trabajadores capacitados y experimentados para operarlos; por ejemplo, en aplicaciones que permiten la inspección de tuberías complejas, donde se requiere una solución de reconstrucción 3D rápida y precisa, para la inspección de la forma externa de las tuberías antes del montaje [Tian *et al.*, 2016].

Con todo lo mencionado anteriormente, elaborar una metodología para examinar distintos tipos de objetos complejos de una manera general, precisa, robusta y adaptable sigue siendo un objetivo todavía no alcanzado. Pero, hay un enfoque que no ha sido contemplado y puede potenciar las técnicas de inspección ya desarrolladas. Con eso nos referimos a implementar, en los procesos de inspección visual existentes, una perspectiva de búsqueda activa. Es decir, realizar una adquisición de múltiples vistas relevantes del objeto desde diferentes ángulos, para luego realizar la reconstrucción 3D y finalmente, un análisis de detección de daño. Si se detecta algo sospechoso, se investiga más a fondo para determinar el área completa del posible daño que podría tener el objeto y calcular su volumen. En caso de no encontrar daño en las vistas iniciales, se continúa con la inspección hasta que todo el objeto esté visualmente cubierto. Creemos que una metodología así puede llegar a ser: *i*) un método innovador para la inspección visual automática de objetos; *ii*) un método de inspección visual eficiente; *iii*) una adecuada fusión de algoritmos de visión por computador y búsqueda activa, aplicados a problemas de inspección; y *iv*) es una metodología robusta y genérica para la inspección de objetos complejos.

¹Entenderemos como daño a la diferencia volumétrica existente al comparar un objeto de prueba con un objeto base. Para este trabajo existirán dos tipos de daño: daño sustractivo y daño aditivo. Por daño sustractivo nos referimos a la diferencia volumétrica negativa resultante de la comparación; en otras palabras, a la ausencia de superficie en el objeto de prueba si se compara con el objeto base. En cambio, daño aditivo se define como la diferencia volumétrica positiva resultante de la comparación y se puede expresar como la presencia de superficie adicional en el objeto de prueba si se compara con el objeto base.

1.1. Hipótesis

Es posible detectar y cuantificar el daño superficial de un objeto utilizando la inspección visual de manera activa en conjunto con la reconstrucción tridimensional.

1.2. Objetivos

A continuación detallamos el objetivo general y los objetivos específicos de nuestra propuesta de investigación, y así verificamos el cumplimiento de nuestra hipótesis.

1.2.1. Objetivo General

Nuestro objetivo general es elaborar una metodología de inspección visual automática de objetos para detectar y cuantificar eventuales daños, fusionando adecuadamente distintas técnicas de visión por computador y búsqueda activa.

1.2.2. Objetivos Específicos

Para lograr el objetivo general, proponemos los siguientes objetivos específicos:

- a) Diseñar e implementar un sistema de adquisición de imágenes utilizando visión estéreo para la inspección visual automática y conceptualizar su modelo geométrico,
- b) Implementar un algoritmo para el movimiento de una plataforma giratoria basado en la planificación de vistas,
- c) Diseñar e implementar un sistema para la reconstrucción 3D de un objeto,
- d) Implementar un algoritmo de detección de daño en la superficie de un objeto reconstruido en 3D,
- e) Implementar un algoritmo de cálculo de volumen del daño detectado,
- f) Evaluar la propuesta de inspección activa de objetos, utilizando índices de desempeño.

1.3. Contribución

Esta tesis generó una contribución científica, la cual fue publicada en journal WoS Science Citation Index Expanded (SCI-EXPANDED), específicamente en el journal "Measurement", de la editorial Elsevier con un factor de impacto $FI = 5.131$ (Q1).

Capítulo 2

Trabajos Relacionados

Los avances en el área de la inspección no destructiva usando visión por computador muestran un potencial creciente debido a los diferentes enfoques que se han generado a lo largo de los años [Czimmermann *et al.*, 2020], en particular, el desarrollo de sistemas que puedan ser robustos, precisos y eficientes al momento de inspeccionar objetos.

Molleda *et al.* [2010] desarrolló un sistema de inspección de calidad en tiempo real para productos metálicos y láminas metálicas planas. Este sistema se basa en la triangulación óptica, utilizando un patrón de líneas láser, que proporciona la posición de la luz proyectada sobre la superficie del objetivo, a través de una geometría sencilla. Las tiras de acero se mueven a través de una banda de paso, donde una cámara con un sensor CMOS (Complementary Metal-Oxide-Semiconductor) capta la proyección de una línea láser, con el fin de reconstruir la superficie en tres dimensiones y así analizar los dos factores geométricos importantes en las tiras metálicas: planitud y anchura. Tres años más tarde, Molleda *et al.* [2013] implementó mejoras en el rendimiento de detección y extracción, incorporando nuevos tipos de procesamiento: uno de alta velocidad pero menos preciso y otro que proporciona alta precisión aumentando los tiempos de cálculo. También se propuso un mecanismo para reducir los efectos sobre el movimiento de las láminas metálicas durante el proceso de fabricación. Este sistema tiene muchas similitudes a lo que proponemos, en particular al uso de la triangulación óptica para captar la silueta del objeto y luego analizar lo obtenido. Sin embargo, su sistema es poco adaptable y requiere componentes muy específicos para su elaboración, como es el caso de la cámara con sensor CMOS.

Un enfoque interesante implementado en un sistema de inspección automatizado fue la incorporación de una estrategia de reescaneo, que permite minimizar el tiempo de operación, elaborada por Kwon *et al.* [2019]. Al ingresar un modelo CAD del objeto a inspeccionar, el sistema comienza escaneando el área completa del objeto y desarrolla un modelo aproximado para calcular una ruta de escaneo. Luego, se realiza la fase de escaneo completo del objeto de prueba para reconstruirlo y así generar un mapa de errores, que indica las áreas que se representan con grandes errores o que no fueron escaneadas. En su última fase, las áreas indicadas como deficientes por el mapa de error se vuelven a escanear para generar un mapa de error mejorado y luego inspeccionar el objeto con mayor confiabilidad. Esta metodología de reescaneo se asemeja bastante con la búsqueda activa que queremos implementar en un sistema de inspección pero con la diferencia de que no necesitamos realizar muchas etapas al inspeccionar a un objeto. Pretendemos simplificar la secuencia de inspección pero sin perjudicar los resultados obtenidos.

En el mismo año, se implementó un sistema de inspección usando visión estereoscópica y triangulación láser para la reconstrucción 3D [Dawda *et al.*, 2019]. Una de sus grandes ventajas radica en no depender de la cantidad de luz ambiental presente en el momento de la inspección. Su implementación es sencilla: dos cámaras estereoscópicas y un emisor láser que incide sobre el objeto para capturar su silueta y luego obtener la información suficiente para reconstruir. Un año después, optimizaron dicho sistema para que fuera capaz de detectar la presencia de daños en el objeto inspeccionado [Dawda y Nguyen, 2020], mostrando una gran precisión en los resultados obtenidos. Este es uno de los trabajos que más llega a parecerse en cuanto a las técnicas seleccionadas para las etapas de adquisición de la superficie del objeto y reconstrucción 3D. Sin embargo, nuestro sistema utiliza ambas técnicas en conjunto a un enfoque de búsqueda activa, otorgando mayor eficiencia en la ejecución durante las etapas mencionadas anteriormente.

Una propuesta bastante similar es la que entrega Güldür Erkal y Hajjar [2020] respecto a la implementación de técnicas de visión por computador para la inspección de superficies. En este trabajo se diseñó un sistema consistente en cámaras de alta precisión integradas con escáneres láser para la captura de datos de superficie. Posteriormente, con la información capturada del objeto, se ejecuta un método de detección de objetos mediante la voxelización de la nube de puntos obtenida del escaneo láser y luego se utilizan algoritmos de esqueletización del vóxel creado, ob-

teniendo una representación del objeto escaneado. Posteriormente, se compara con un modelo ya definido en una base de datos. Ese análisis de comparación se basa en la alineación de ambos esqueletos, es decir, el del objeto analizado y el perteneciente a la base de datos. Una vez analizado, y si se encuentra algún daño, se utiliza la voxelización ya creada para extraer el valor dimensional del daño detectado. La principal diferencia con nuestro método es que es un método estático, diseñado para la detección y cuantificación de daños en grandes superficies.

Una propuesta interesante para la industria agrícola, generada por Zhang *et al.* [2020], muestra un sistema de visión artificial para la estimación del volumen y peso de manzanas mediante reconstrucción 3D y métodos de medición sin contacto. La superficie 3D de las manzanas fueron reconstruidas utilizando una sola cámara multispectral y una luz estructurada de matriz lineal. Tanto la característica de imagen tradicional como la información de altura se extrajeron de los mapas de altura. Se extrajeron dos tipos diferentes de características de altura y ambas se fusionaron con un área de proyección para formar características combinadas. El sistema y método desarrollado en este estudio proporciona una alternativa a los métodos tradicionales para la medición no contractual del volumen y peso de productos agrícolas. Este trabajo realiza la reconstrucción 3D de forma similar a nuestra propuesta y en lugar de utilizar una plataforma giratoria, utiliza una cinta transportadora. Sin embargo, no se centra en la detección y cuantificación automática de daños.

En el año 2021, se implementó una metodología para la detección de frutas dañadas mediante la medición de ondas milimétricas (mmwave) [Zidane *et al.*, 2021]. Ésta comienza detectando la diferencia de permitividad relativa incluida en las imágenes de ondas milimétricas entre frutos sanos y dañados. Posteriormente se utiliza el algoritmo Support Vector Machine (SVM), el cual es un método de Machine Learning capaz de clasificar los frutos sanos de los dañados. El sistema que se implementó para validar la metodología consta de un escáner 3D esférico, una torre Rohacell donde se coloca la fruta y una antena sonda que gira sobre la fruta con un radio de 585 mm. Para minimizar el tiempo de medición, que alcanzó las 2 horas, y la cantidad de información (número de puntos de medición) utilizada, se implementó el método Grey Wolf Optimizer (GWO) con el fin de encontrar los valores óptimos necesarios para el aprendizaje del algoritmo SVM. En los resultados de su investigación, obtuvieron una precisión entre el 93 % y 100 % en las pruebas realizadas con melocotones y manzanas, tanto dañados como sanos. Este trabajo muestra resulta-

dos preliminares y no concluyentes ya que es un modelo de inspección estático para la detección de daños en frutos y no muestra un método para la cuantificación de los daños detectados. Es un trabajo diferente al nuestro ya que utilizamos diferentes algoritmos y medios para obtener información.

Liu *et al.* [2021] propuso un método basado en técnicas de reconstrucción de daños superficiales estructurales en 3D para reconstruir y extraer datos para el cálculo de volumen del daño. El daño superficial de la muestra de concreto se reconstruye tridimensionalmente utilizando imágenes de vistas múltiples tomadas con teléfonos inteligentes y comparadas con una cámara de profundidad. Se obtuvieron los datos de la nube de puntos, y luego se ajustó y eliminó el plano de daño mediante un algoritmo de Consenso de Muestra Aleatoria para obtener los datos del sitio del daño; finalmente, se calculó el volumen de daño. Para determinar el volumen de daño se hace la diferencia entre el volumen reconstruido con la superficie plana sin daño de manera similar a nuestra propuesta. En esta propuesta no se considera ni la inspección activa ni la detección automática del área dañada, y en consecuencia discriminar si hay -o no- daño. Además, nuestro método considera la inspección de objetos con formas irregulares y realizamos una inspección completa de dichos objetos.

Durante el mismo año, Guan *et al.* [2021] desarrolló un marco de trabajo automatizado propuesto para la detección de fallas en el pavimento a nivel de píxel utilizando de manera conjunta la visión estéreo y un aprendizaje profundo basado en U-net, el cuál es una red neuronal que se caracteriza por poseer una conexión de acceso directo, que integra varios niveles de abstracción. Todo comienza con la adquisición de imágenes de alta resolución mediante fotografía paralela y fotografía oblicua. Luego, se crea un modelo de nube de puntos del pavimento en 3D utilizando visión estéreo. Posteriormente, mediante el uso de un algoritmo de procesamiento de nube de puntos con el modelo 3D del pavimento, se obtienen conjuntos de datos de imágenes de características múltiples que incluyen imágenes en color, imágenes de profundidad e imágenes superpuestas de profundidad de color. Luego, se lleva a cabo la detección de las grietas y baches presentes en el pavimento mediante una segmentación utilizando aprendizaje profundo. Por último, se realiza una medición automatizada del volumen de los baches en el pavimento a través de un algoritmo de medición de volumen implementado. Consiguieron, con sus pruebas, una precisión de un 0,9632. Cabe destacar que, pese a que lograron realizar una metodología cuyos

objetivos son idénticos al que queremos conseguir y con unos valores de desempeño muy buenos, una de las ventajas que tiene nuestra metodología es que pueda ser fácilmente adaptable.

Más recientemente, Shim *et al.* [2022] propuso un método de detección de fisuras en el hormigón utilizando una red neuronal profunda que aplica una proporción de mezcla óptima de datos de entrenamiento clasificados en tres grupos: O-Images, N-Images y S-Images para mejorar el rendimiento del reconocimiento de las grietas. Además, añadieron una red discriminadora de conjuntos equilibrados basada en el aprendizaje contradictorio. Este último es un método de aprendizaje semisupervizado que se caracteriza por la aplicación simultánea de dos conjuntos de datos de aprendizaje: los etiquetados y los no etiquetados. La red neuronal profunda utilizada comprende de una red neuronal convolucional, lo que le permite detectar grietas multiescala. Además, incorporaron un método de reconstrucción tridimensional de las grietas utilizando una técnica de medición de triangulación basada en visión estereoscópica que determina el tamaño de las grietas detectadas. Los resultados experimentales que realizaron demostraron que el algoritmo propuesto logró un rendimiento de detección con una intersección sobre unión media del 84,53% y una puntuación F1 del 82,91%. A pesar de que este método es capaz de detectar de manera bastante eficaz el daño, grietas para este caso, que puede presentar el hormigón, no logra igualar nuestra propuesta debido a la carencia de un método para cuantificar el daño detectado.

Después de revisar la literatura y trabajos relacionados, podemos decir que la reconstrucción 3D está muy desarrollada y se utilizan diferentes estrategias para llevarla a cabo. Las dos estrategias principales son el uso de vistas múltiples y la proyección de luz estructurada. Además, hemos visto que la detección de daños sigue siendo una cuestión abierta, con una amplia variedad de algoritmos y estrategias que van desde el procesamiento de imágenes hasta el uso de redes neuronales convolucionales (CNN). Debido a las dificultades que implica el cálculo del volumen, pocos trabajos se centran en este problema. No hemos encontrado propuestas en la literatura que combinen de manera eficiente estrategias que permitan realizar una inspección visual automatizada de los objetos para buscar activamente algún daño en ellos (como lo haría un humano), y que incorporen la cuantificación del daño. Creemos que nuestra propuesta es innovadora, genérica, fácilmente adaptable y brinda resultados auspiciosos.

Capítulo 3

Método Propuesto

A continuación abordaremos la metodología que utilizaremos para la detección y cuantificación de daño en un objeto.

3.1. Descripción de la metodología

La metodología utilizada en esta propuesta se basa en una modificación de Knowledge Discovery in Databases (KDD). Esta es una metodología bien conocida para la extracción automatizada de conocimiento a partir de grandes volúmenes de datos y está compuesta por cinco etapas; Adquisición y selección de datos, Pre-procesamiento, Transformación, Minería de datos y Evaluación e interpretación [Mitra *et al.*, 2002].

- Adquisición y selección de datos: Como su nombre lo indica, se realiza una recopilación de la información existente de todas las fuentes disponibles. Posteriormente se analiza todo lo obtenido y se eligen los datos más relevantes.
- Pre-procesamiento: Esta etapa se enfoca en mejorar la calidad de los datos utilizando operaciones como eliminación de ruido y manejo de datos faltantes.
- Transformación: Su objetivo es transformar los datos pre-procesados en formas apropiadas para la siguiente etapa, es decir, encontrar características útiles para representar los datos debido a que se aplicarán una serie de algoritmos sobre estos.

- Minería de datos: Es la etapa más representativa del proceso KDD y es habitual utilizarla para referirse a todo el proceso. En ella se emplean métodos inteligentes para extraer patrones de datos.
- Evaluación e interpretación: De los patrones obtenidos en la fase anterior, se efectúan análisis para ser evaluados e interpretados para así convertirse en conocimiento.

Debido a que esta metodología es iterativa, permite una mejor interpretación de lo adquirido.

Para efectos de nuestra propuesta, la metodología modificada nos queda constituida por las siguientes etapas:

- Adquisición de Imágenes: Toda la información necesaria para la detección de daño en un objeto se encuentra en su superficie, por lo que es importante escoger un método de inspección adecuado. De ello depende el buen desarrollo de las etapas posteriores. Después de un análisis, que será explicado en la sección 3.1, se ha decidido utilizar la metodología visual para extraer la superficie del objeto. Por lo tanto, esta etapa se enfoca en la obtención de capturas a la superficie del objeto de prueba.
- Pre-procesamiento de Imágenes: Para realizar una correcta reconstrucción 3D, es necesario mejorar la calidad de las imágenes, mediante un pre-procesamiento que sea útil en el algoritmo. Debido a que las imágenes adquiridas pueden tener diferencias de iluminación y sombras, será necesaria la aplicación de algoritmos de eliminación de ruido, mejora de contraste, segmentación, entre otros.
- Reconstrucción 3D: Esta etapa es posible mediante el uso de la matriz de proyección de cámara. Para ello previamente se deben calibrar las cámaras, utilizando un patrón de calibración para obtener los parámetros extrínsecos e intrínsecos de éstas. Con dichos valores, transformamos cada punto 2D perteneciente a las imágenes pre-procesadas en un punto 3D, perteneciente al mundo. Como resultado obtenemos una nube de puntos que reconstruye a la región analizada del objeto de prueba.
- Alineación de nubes de puntos 3D: Se establecen las correspondencias entre la región reconstruida 3D del objeto de prueba y un modelo ideal del mismo que se

encuentra almacenado en una base de datos. Para ello será necesario utilizar un descriptor 3D que pueda aprovechar toda la información disponible de las etapas anteriores para poder alinear dichas nubes de puntos.

- **Detección de Daño:** Una vez obtenida la reconstrucción del objeto de prueba, se realizará un análisis comparativo de superficies entre la reconstrucción 3D del objeto de prueba y una versión perfecta del mismo la cuál está almacenada en una base de datos. Todo eso con la finalidad de detectar la presencia de algún daño o factor de baja calidad. En caso de haber encontrado la existencia de daño, se generará una nube de puntos donde estará alojada solamente el área dañada.
- **Cálculo de Volumen:** Con la nube de puntos del área dañada, se calcula el volumen mediante el uso de *AlphaShape*, que permite la creación de polígonos que envuelven todos los puntos que conforman la nube.

En el algoritmo 1 se puede apreciar de manera general la metodología propuesta anteriormente. En este algoritmo, los parámetros *lim1* y *lim2* son los límites de *Dmg_Obj*. Estos valores se calculan encontrando primero los puntos límite, tanto a la izquierda como a la derecha, y luego determinando en qué paso se obtuvieron durante la etapa de adquisición de la imagen. Estos parámetros, *lim1* y *lim2*, se utilizan para determinar si *Dmg_Obj* está incompleto y en qué dirección de rotación se realizará la nueva iteración del análisis del objeto para completar el área dañada. Esto se puede definir de la siguiente manera: si *lim1* es igual a *IStep* significa que el área dañada está incompleta en el sector izquierdo y por lo tanto, se debe realizar un nuevo análisis girando en sentido antihorario. Por otro lado, si *lim2* es igual a *FStep*, significa que el área dañada está incompleta en el lado derecho y el objeto debe analizarse en sentido horario. Si *lim1* es diferente de *IStep* y a su vez *lim2* es diferente de *FStep*, significa que los límites de *Dmg_Obj* no están cerca de los pasos realizados durante el análisis y el daño es completamente definido.

Es entonces un proceso de inspección activa, que comienza con el objeto de prueba dispuesto en diferentes posiciones iniciales, y luego busca activamente el daño, provocando rotaciones en sentido horario o antihorario como se describió anteriormente, hasta que el daño está completamente delimitado y se puede calcular (ver resultados de este trabajo en la Tabla 4.1).

Algoritmo 1: Metodología de Inspección

Entrada: $IStep = 1$ %Paso inicial, $FStep = 50$ %Paso final,
 $dir = 1$ %Dirección de rotación, $ncaps = 50$ %número de capturas
realizadas, P %Matriz de proyección, $TObj$ %Reconstrucción 3D
del objeto de prueba.

Salida: Vol %Volumen de daño.

```
1 mientras  $ncaps < 200$  hacer
2   %Etapa 1: Adquisición de imágenes
3   [ $LOn, LOff$ ]  $\leftarrow$  Adquirir_Img( $IStep, FStep, dir$ );
4   %Etapa 2: Pre-procesamiento
5   [ $Img\_Skel, \_$ ]  $\leftarrow$  Pre-proc( $LOn, LOff$ );
6   %Etapa 3: Reconstrucción 3D
7   [ $TObj, RGB\_TObj$ ]  $\leftarrow$  Reconstrucción_3D( $dir, Img\_Skel, P$ );
8   %Etapa 4: Alineación de nubes de puntos 3D
9    $Align\_SObj$   $\leftarrow$  Alineación( $TObj, SObj$ );
10  %Etapa 5: Detección de daño
11   $Dmg\_Obj$   $\leftarrow$  Detección( $Align\_SObj, TObj$ );
12  si  $Dmg\_Obj \neq 0$  entonces
13    si  $lim1 = IStep$  entonces
14      |  $IStep = FStep + 1$ ;
15      |  $FStep = FStep + 15$ ;
16    en otro caso
17      si  $lim2 = FStep$  entonces
18        |  $IStep = FStep + 1$ ;
19        |  $FStep = FStep + 15$ ;
20        |  $dir = -1$ ;
21      en otro caso
22        | %Etapa 6: Cálculo de volumen
23        |  $Vol \leftarrow$  Calc_Vol( $Dmg\_Obj$ );
24      fin
25    fin
26     $ncaps = ncaps + 15$ ;
27  en otro caso
28    |  $IStep = FStep + 1$ ;
29    |  $FStep = FStep + 30$ ;
30  fin
31 fin
```

3.2. Método de Inspección

Escoger la método de inspección adecuado es indispensable para el desarrollo de nuestra metodología fundamental, ya que de esta elección determinaremos el mecanismo de reconstrucción tridimensional del objeto.

En general, existen dos conjuntos de inspección: *i*) los de contacto directo con el objeto, mediante una punta de prueba que recorre su superficie, transfiriendo las coordenadas de cada punto al computador. Este sistema de desplazamiento de la punta de prueba otorga una gran resolución al modelo reconstruido, pero resulta ser muy lento y eventualmente podría ser perjudicial para el objeto en estudio, debido a que el contacto con la punta de prueba puede dañar su superficie [Kus, 2009]; *ii*) eliminando la necesidad de contacto directo con el objeto, enfocando su interacción en otros aspectos, por ejemplo, interacción óptica. Estas son denominadas Pruebas No Destructivas (NDT) y son utilizadas en diferentes áreas debido a que se pueden realizar durante cualquiera de las etapas de vida útil del objeto [Gupta *et al.*, 2021]. Este último conjunto está más enfocado a la evaluación y detección de daños superficiales y por ello es la opción más adecuada debido a la simplicidad a la hora de implementar y manipular.

3.3. Adquisición de Imágenes

Con el método de inspección elegido, necesitamos profundizar en la técnica de adquisición de imágenes que nos permita transformar una imagen en un conjunto de datos que puedan ser manipulados [Mishra *et al.*, 2017]. Para nuestro caso, esos datos nos servirán para generar una reconstrucción tridimensional del objeto.

La técnica de triangulación activa permite determinar la topografía 3D del objeto de prueba utilizando un principio geométrico: rayos de luz, que pueden ser emitidos por un proyector de vídeo digital, un proyector de diapositivas analógico o un láser [Drouin y Beraldin, 2020] que incide sobre la superficie del objeto de prueba [Cock, 2012]. Hemos decidido utilizar esta técnica ya que, para un sistema de inspección visual automático, es relativamente simple de implementar [Mayorca y López Rubio, 2018]. Aparte, se ha decidido incorporar visión estéreo.

El sistema propuesto para una adecuada adquisición de imágenes del objeto de prueba requiere: una plataforma giratoria, compuesta por un motor paso a paso que permita un control de movimiento; dos cámaras en configuración estéreo, con

la posibilidad de poder configurar su posición, para así modificar el punto de vista; uno o dos emisores de línea láser y un computador, que será el encargado de realizar el proceso de adquisición y almacenaje de las imágenes (ver Figura 3.1).

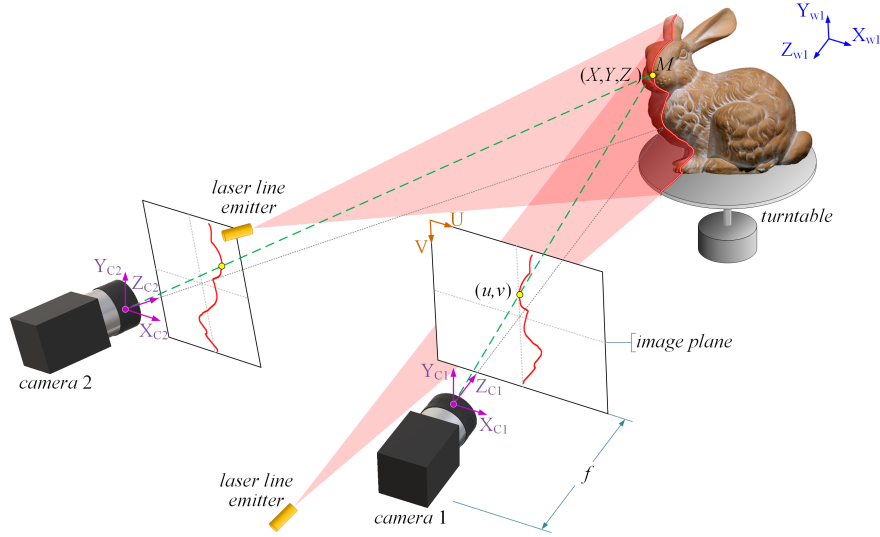


Figura 3.1: Sistema de inspección para la detección de un eventual daño.

Para poder realizar la reconstrucción 3D de los objetos dispuestos en la plataforma giratoria, es necesario que previamente al proceso de inspección, las cámaras sean calibradas. Este proceso permitirá obtener los parámetros intrínsecos y extrínsecos de las cámaras. Los parámetros intrínsecos, indican las coordenadas de un punto 2D (pixel) en la imagen a partir de la posición del punto 3D en la escena y son; la distancia focal, el centro óptico y los coeficientes de distorsión [Long y Dongri, 2019]. Los parámetros extrínsecos miden la posición y orientación de la cámara respecto al sistema de coordenadas establecido para el mundo, y son; la rotación y traslación.

De los métodos de calibración existentes, el método de Zhang [2000] se toma como referencia. Este método utiliza las coordenadas de los puntos situados en un patrón de calibración, similar a un tablero de ajedrez. Las imágenes del patrón de calibración son adquiridas considerando distintas posiciones y orientaciones.

Una vez definida la estructura de la adquisición de imágenes, se definen los parámetros $IStep$ y $FStep$ como el paso inicial y final de la plataforma giratoria, respectivamente, y dir como la dirección de rotación. La adquisición de imágenes del objeto se realiza usando la secuencia de acciones mostrada en el algoritmo 2.

Algoritmo 2: Función `Adquirir_Img`

Entrada: $IStep = 1$ %Paso inicial, $FStep = 50$ %Paso final,
 $dir = 1$ %Dirección de rotación.

Salida: LOn %Vector que contiene las capturas del objeto con línea láser proyectada, $LOff$ %Vector que contiene las capturas del objeto sin línea láser proyectada.

```
1 Función [ $LOn, LOff$ ]  $\leftarrow$  Adquirir_Img(IStep, FStep, dir)
2   para  $n = IStep$  a  $FStep$  hacer
3     Rotar la plataforma  $n$  pasos en la dirección  $dir$ ;
4     Encender el diodo láser;
5      $LOn(n, :)$   $\leftarrow$  Capturar imagen;
6     Apagar el diodo láser;
7      $LOff(n, :)$   $\leftarrow$  Capturar imagen;
8   fin
9 fin
```

3.4. Pre-procesamiento de Imágenes

Es necesario depurar las imágenes obtenidas antes de realizar la reconstrucción 3D para evitar que cualquier factor no deseado presente en ellas, como diferencias de iluminación, sombras, o distorsión generada por el lente de la cámara [Gonzalez y Woods, 2018; Tyagi, 2018], puedan generar inconvenientes en etapas posteriores.

Para corregir la presencia de distorsión radial en cada imagen L_On y L_Off (con láser encendido y apagado, respectivamente), provocada por el lente de la cámara, utilizamos los coeficientes de distorsión radial, obtenidos del proceso de calibración de cámara. Para obtener de cada píxel localizado en la imagen, su equivalente no distorsionado, despejamos las coordenadas (x, y) desde las ecuaciones (3.1) y (3.2) respectivamente:

$$\check{x} = x(1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6), \quad (3.1)$$

$$\check{y} = y(1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6), \quad (3.2)$$

donde, (\check{x}, \check{y}) y (x, y) son las coordenadas distorsionadas y no distorsionadas, respectivamente, del píxel, k_1 , k_2 y k_3 son los coeficientes de distorsión radial y $r = \sqrt{x^2 + y^2}$.

De la acción anterior obtenemos el conjunto de imágenes no distorsionadas

ND_LOn y ND_LOff . Para cada par de imágenes asociadas a un paso determinado.

Luego de eliminar la distorsión radial producida por el lente de la cámara y obtener ND_LOn y ND_LOff , es necesario segmentar la línea láser, para lo cual se extraen desde las imágenes ND_LOn y ND_LOff sus respectivos canales rojos (R_LOn y R_LOff) y luego se restan, obteniendo como resultado una imagen donde solo se visualiza la línea láser proyectada, denominada Img_Laser , como se muestra en la Figura 3.2. Una vez segmentada la línea láser, se requiere obtener el esqueleto de dicha línea, para lo cual, se analiza cada píxel de Img_Laser en búsqueda del valor de intensidad máximo presente, v_max . Con este dato calculamos el valor mínimo de intensidad: $v_min = 0,85 \cdot v_max$, donde, el escalar 0.85 fue determinado empíricamente. Con estos valores de v_min y v_max , calculamos el valor umbral U adecuado para binarizar Img_Laser , usando: $U = v_max - v_min$.

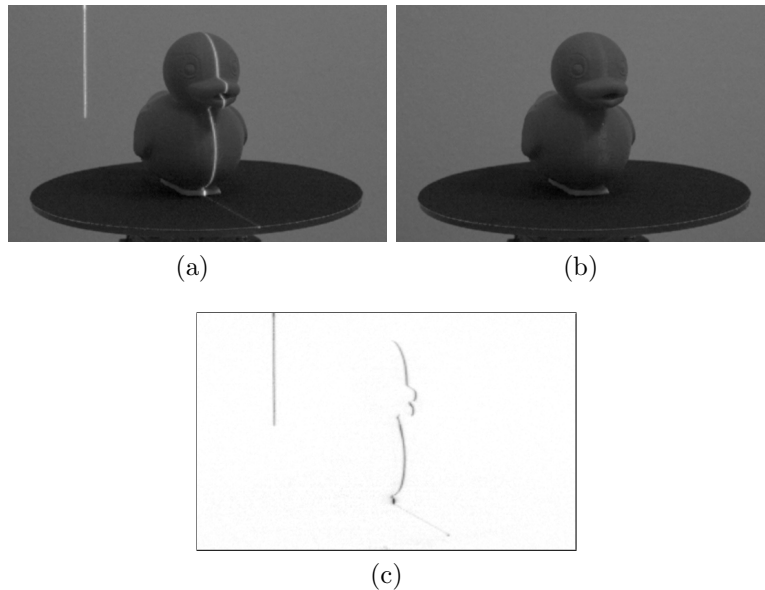


Figura 3.2: Segmentación de la línea láser. (a) Canal rojo de ND_LOn , (b) Canal rojo de ND_LOff , (c) Resta de canales de rojo ($R_LOn - R_LOff$).

Con Img_Laser binarizado, aplicamos una erosión utilizando un disco de 1 mm de radio como objeto del elemento estructurante, y así finalmente se realiza la esquelización. Para obtener el esqueleto de la línea láser proyectada sobre el objeto, generamos una región de interés en la imagen, para determinar la ubicación de la línea y descartar el entorno de la escena. El resultado del proceso de esquelización

y los pasos previos, se pueden ver en la Figura 3.3.

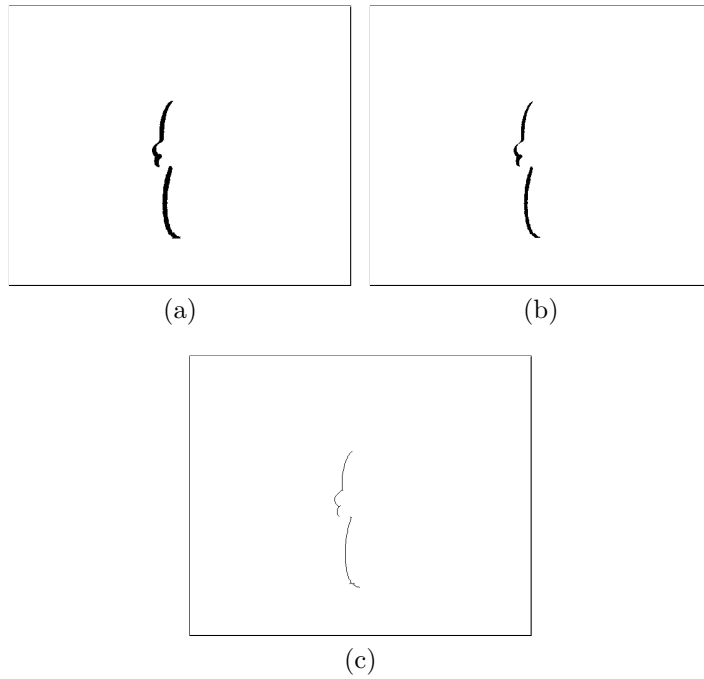


Figura 3.3: Esqueletización de la línea segmentada. (a) Binarización de *Img_Laser*, (b) Erosión utilizando un disco de 1 mm de radio, y (c) Esqueletización.

Para la etapa de detección de daño, es importante identificar cada píxel del esqueleto de la línea láser en la imagen *ND_LOff*, con el propósito de almacenar el conjunto de colores *RGB* (*Red*, *Green*, *Blue*) respectivo de la superficie del objeto en la que fue proyecta la línea láser. Todo esto queda resumido mediante el algoritmo 3.

Algoritmo 3: Función Pre-proc

Entrada: $ncaps = 50$ %Número de capturas realizadas, LOn %Vector que contiene las capturas del objeto con línea láser proyectada, $LOff$ %Vector que contiene las capturas del objeto sin línea láser proyectada.

Salida: Img_Skel %Vector que contiene las líneas láser esqueletizadas, RGB_Skel %Vector que contiene los valores RGB de las líneas láser esqueletizadas.

```
1 Función [ $Img\_Skel, RGB\_Skel$ ]  $\leftarrow$  Pre-proc( $LOn, LOff, ncaps$ )
2   para  $n = 1$  to  $ncaps$  hacer
3      $ND\_LOn(n, :)$   $\leftarrow$  Eliminar la distorsión de  $LOn(n, :)$ ;
4      $ND\_LOff(n, :)$   $\leftarrow$  Eliminar la distorsión de  $LOff(n, :)$ ;
5      $R\_LOn(n, :)$   $\leftarrow$  Extraer el canal rojo de  $ND\_LOn(n, :)$ ;
6      $R\_LOff(n, :)$   $\leftarrow$  Extraer el canal rojo de  $ND\_LOff(n, :)$ ;
7      $Img\_Laser(n, :)$   $\leftarrow R\_LOn(n, :) - R\_LOff(n, :)$ ;
8      $v\_max$   $\leftarrow$  Calcular el valor de intensidad máximo en
        $Img\_Laser(n, :)$ ;
9      $v\_min$   $\leftarrow 0,85 \cdot v\_max$ ;
10     $U$   $\leftarrow v\_max - v\_min$ ;
11     $Img\_Bin(n, :)$   $\leftarrow$  Binarizar  $Img\_Laser(n, :)$  bajo el umbral  $U$ ;
12     $Img\_Bin.2(n, :)$   $\leftarrow$  Realizar erosión usando un disco de 1 mm de
       radio a  $Img\_Bin(n, :)$ ;
13     $Img\_Skel(n, :)$   $\leftarrow$  Esqueletizar  $Img\_Bin.2(n, :)$ ;
14     $RGB\_Skel(n, :)$   $\leftarrow$  Extraer valores RGB de  $LOff(n, :)$ ;
15  fin
16 fin
```

3.5. Reconstrucción 3D

Para efectuar la reconstrucción 3D del esqueleto de la línea láser, formada por los puntos (u, v) , es decir, realizar la transformación $(u, v) \mapsto (X, Y, Z)$, usaremos el modelo geométrico mostrado en la Figura 3.1. Solo explicaremos el procedimiento que permite la reconstrucción 3D de las imágenes adquiridas con la cámara 1, ya que, para la cámara 2 el procedimiento es exactamente el mismo.

Comenzaremos explicando el modelo geométrico, en donde inicialmente se observa la relación de un punto 3D y su proyección en una imagen como un punto 2D (mapeo 3D \mapsto 2D), lo cual puede ser expresado como un mapeo lineal en coordenadas homogéneas:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.3)$$

donde, λ es un factor de escala, P es la matriz de proyección de 3×4 , los puntos (u, v) y (X, Y, Z) son las coordenadas 2D y 3D respectivamente, de cada punto de la línea láser.

La matriz de proyección P , puede ser expresada como:

$$P = \begin{bmatrix} \alpha_x & s & u_o \\ 0 & \alpha_y & v_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_w & t_w \\ 0^\top & 1 \end{bmatrix} \quad (3.4)$$

donde, α_x y α_y son los factores de escala, s es el factor de torsión skew, (u_o, v_o) es el punto principal 2D de la imagen en píxeles (usualmente el centro de la imagen) y f es la distancia focal. A estos parámetros se les denomina *parámetros intrínsecos*. Además, los *parámetros extrínsecos* son: R_w que es la matriz de rotación de 3×3 , y t_w es el vector traslación de 3×1 , ambos, R_w y t_w respecto a las coordenadas del mundo (X_{w1}, Y_{w1}, Z_{w1}) . Tanto los parámetros extrínsecos e intrínsecos pueden ser estimados mediante un proceso de calibración [Zhang, 2000].

Dado que P es una matriz de 3×4 , la ecuación (3.3) puede ser escrita como:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.5)$$

Resolviendo esta ecuación, se obtiene el siguiente sistema de ecuaciones simultáneas:

$$\begin{aligned} \lambda u &= P_{11}X + P_{12}Y + P_{13}Z + P_{14} \\ \lambda v &= P_{21}X + P_{22}Y + P_{23}Z + P_{24} \\ \lambda &= P_{31}X + P_{32}Y + P_{33}Z + P_{34} \end{aligned} \quad (3.6)$$

Reemplazando la tercera ecuación de (3.6) en las dos primeras, y ordenando, podemos obtener las matrices A y B del sistema de ecuaciones lineales $Ax = B$. De esta forma A y B serán:

$$A = \begin{bmatrix} (uP_{31} - P_{11}) & (uP_{32} - P_{12}) & (uP_{33} - P_{13}) \\ (vP_{31} - P_{21}) & (vP_{32} - P_{22}) & (vP_{33} - P_{23}) \end{bmatrix} \quad (3.7)$$

$$B = \begin{bmatrix} -uP_{34} + P_{14} \\ -vP_{34} + P_{24} \end{bmatrix} \quad (3.8)$$

Utilizando A y B obtenemos los puntos 3D (X, Y, Z) de la línea láser, con respecto al sistema de coordenadas del mundo (X_{w1}, Y_{w1}, Z_{w1}) , tal como sigue:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A \setminus B \quad (3.9)$$

Sin embargo, como el objeto inspeccionado está sobre una plataforma giratoria, se debe considerar dicha rotación por cada adquisición e incorporar el desplazamiento a los puntos (X, Y, Z) . Por tal motivo, debemos incorporar una nueva transformación $(X, Y, Z) \mapsto (X_p, Y_p, Z_p)$, como sigue:

$$\begin{bmatrix} X_p & Y_p & Z_p \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^T \cdot R_p + t_p \quad (3.10)$$

Como la plataforma giratoria solo genera movimiento de rotación, se considera que el vector de traslación $t_p = 0$, quedando:

$$\begin{bmatrix} X_p & Y_p & Z_p \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^T \cdot R_p \quad (3.11)$$

donde R_p es la matriz de rotación, definida como:

$$R_p = R_{px}(\alpha) \cdot R_{py}(\beta) \cdot R_{pz}(\gamma) \quad (3.12)$$

donde, R_{px} , R_{py} y R_{pz} son las matrices de rotación en torno a cada eje de coordenadas de la plataforma giratoria, dado los ángulos α , β y γ respectivamente. Las ecuaciones para calcular cada matriz son las siguientes:

$$R_{px}(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad (3.13)$$

$$R_{py}(\beta) = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \quad (3.14)$$

$$R_{pz}(\gamma) = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.15)$$

Dado que la rotación solo se realiza para el eje Y , es decir, $\alpha = 0$ y $\gamma = 0$, por lo que el ángulo de rotación β es una variable que debe ser calculada como:

$$\beta = stp \cdot (1,8 \cdot dir) \quad (3.16)$$

donde, stp indica el número del paso dado al adquirir la imagen ($stp = 1, 2, \dots, 200$) y dir es el sentido de la rotación dada por la plataforma giratoria, si la rotación es en sentido horario $dir = 1$, de lo contrario, para sentido anti-horario $dir = -1$.

Una vez obtenida la matriz de rotación R_p , reemplazamos este valor en la ecuación (3.11), para finalmente obtener el conjunto de puntos 3D (X_p, Y_p, Z_p) que definen a la región del objeto que está siendo inspeccionada.

3.6. Alineación de Nubes de Puntos 3D

La alineación de nubes de puntos 3D, permite establecer las correspondencias entre las imágenes de regiones adquiridas del objeto de prueba y el modelo ideal del mismo, que se encuentra almacenado en una base de datos.

Para esta alineación, necesitamos utilizar un descriptor 3D que pueda aprovechar de manera eficiente toda la información extraída de las etapas de adquisición y reconstrucción 3D. El *descriptor de covarianza multiescala* (MCOV) [Cirujeda *et al.*, 2015] implementa mejoras en el cálculo de su desempeño, mediante la incorporación de varias magnitudes de vecindad, para cada punto central que se analiza. Además, este descriptor se complementa con un enfoque de teoría de juegos, llamado estrategia evolutiva estable (ESS), para resolver las correspondencias coincidentes bajo restricciones geométricas globales. Esta capa ofrece una comprensión completa de la escena y evita posibles desajustes debido a áreas repetidas o simetrías, que serían imposibles de identificar únicamente por un detector a nivel local.

Para seleccionar aquellos puntos que puedan describir al objeto, este descriptor utiliza matrices de covarianza, cuyas dimensiones dependen directamente del número de características utilizadas. El descriptor usa 6 características, que pueden ser extraídas de una nube de puntos; las medidas angulares invariantes rotacionales α , β y γ junto con los valores del espacio de color R , G y B .

Posteriormente, se calcula la matriz de distancias entre los descriptores, para así obtener un conjunto de pares candidatos. Usando el solucionador ESS, se descartan los candidatos obtenidos anteriormente, hasta encontrar los mejores pares a través de una función heurística definida. Finalmente, con los pares clasificados, se obtienen las transformaciones rígidas (matriz de rotación R y vector de traslación t) para alinear las nubes de puntos. Ver más detalle teóricos del descriptor MCOV en [Cirujeda *et al.*, 2015].

La etapa de alineación se define de la siguiente manera: Antes de comenzar y como requisito previo, es necesario unificar las coordenadas de cada punto con sus respectivos valores RGB . Luego, como entradas se usa la región adquirida del objeto de prueba, (Target Object: $TObj$), y el modelo generado por una versión ideal del mismo, el cual denominaremos objeto base (Source Object: $SObj$), con el fin de obtener la matriz de rotación, R y el vector de traslación t , para efectuar la transformación del objeto base y alinearlo con respecto al objeto de prueba, como se aprecia en la figura 3.4.

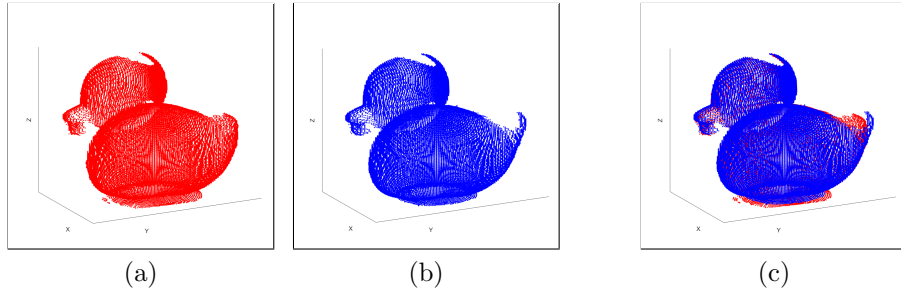


Figura 3.4: Alineación de Objetos. (a) Objeto base, $SObj$, (b) Objeto de prueba, $TObj$, (c) Objeto base y de prueba alineados.

Algoritmo 4: Función Alineación

Entrada: $TObj$ %Reconstrucción 3D Reconstruction del objeto de prueba,
 $SObj$ %Reconstrucción 3D del objeto base.

Salida: Al_SObj %Reconstrucción 3D alineada del objeto base.

```

1 Función  $Al\_SObj \leftarrow$  Alineación( $TObj, SObj$ )
2   |  $[R, t] \leftarrow$  Descriptor_MCOV( $TObj, SObj$ );
3   |  $Al\_SObj \leftarrow$  Alinear  $SObj$  con  $TObj$  usando  $R$  y  $t$ ;
4 fin

```

3.7. Detección de Daño

Con ambas nubes de puntos alineadas ($SObj$ y $TObj$), podemos iniciar la detección de daño. Esta etapa es la más importante en nuestra metodología debido a que es aquí en donde se realiza la detección de la existencia -o no- de alguna diferencia, mediante la comparación entre $SObj$ y $TObj$. En caso de existir, denominamos como “daño” presente en el objeto de prueba.

Dicho esto, es necesario detallar el procedimiento definido para cumplir con el propósito de esta etapa; Comenzamos realizando una búsqueda de todos los pares de puntos equivalentes existentes entre $TObj$ y $SObj$. Debido a que las dos reconstrucciones 3D pertenecen a un mismo tipo de objeto, se puede deducir que todos los puntos entre ambas nubes de puntos tienen su semejante en términos de ubicación espacial. En otras palabras, cada punto perteneciente a $TObj$ puede ser asociado a un punto en $SObj$ en términos de coordenadas X , Y y Z .

Para ello, utilizamos el clasificador, k -NN, es decir, k vecino más cercano [Wang *et al.*, 2003], el cual se ejecuta de la siguiente forma; Sea p_t un punto perteneciente a $TObj$ y p_s un punto perteneciente a $SObj$, entonces para calcular el k vecino más cercano de p_t en relación a $SObj$ se debe realizar un cálculo iterativo de la distancia euclídeana [Phillips, 2021] entre p_t y todos los puntos p_s que conforman $SObj$. El cálculo de la distancia euclídeana d entre dos puntos p_t y p_s se realiza de la siguiente manera:

$$d(p_t, p_s) = \sqrt{(x_t - x_s)^2 + (y_t - y_s)^2 + (z_t - z_s)^2} \quad (3.17)$$

donde, x_t , y_t y z_t son las coordenadas de un punto p_t perteneciente a $TObj$, y x_s , y_s y z_s son las coordenadas de un punto p_s ubicado en $SObj$, ambos con respecto a los ejes X , Y y Z .

Al obtener las distancias d entre p_t y todos los puntos p_s que conforman $SObj$, se seleccionan los k puntos con los menores valores, siendo k , el número de puntos vecinos que se deseen encontrar. Para efectos de nuestro método, hemos ajustado $k = 1$.

Como resultado, obtenemos todos los pares de puntos equivalentes entre $SObj$ y $TObj$, los cuales han sido almacenados en un vector llamado eq_pts .

Para determinar si los pares de puntos equivalentes en eq_pts pertenecen a una región de daño, es necesario definir el criterio, que indique si un par de puntos equivalentes se considera como “daño” -o no-. Para esto, utilizamos los valores de distancia d obtenidos anteriormente y lo comparamos con un valor umbral empírico $minDist$, el cual se define como la distancia límite mínima para determinar cuando un par de puntos equivalentes puede ser considerado como “daño”. Todo esto está establecido bajo la siguiente condición:

Si $d \geq minDist$, el par de puntos equivalentes *califica* como “daño” presente en la superficie del objeto de prueba y es almacenado en el vector Dmg_Obj . De lo

contrario, sí $d < minDist$, el par de puntos equivalentes *no califica* y es descartado.

Como eventualmente podría haber valores que sobrepasen el criterio del valor umbral $minDist$, aplicamos un filtro para eliminar cualquier par de puntos equivalentes que no pertenezcan verdaderamente a Dmg_Obj . Esto se logra analizando la distribución de los puntos que conforman Dmg_Obj , es decir, buscamos el número de paso stp de cada punto en Dmg_Obj y luego analizamos la frecuencia de pasos, el cuál obtuvimos en la etapa de reconstrucción 3D.

Con el procedimiento descrito, el cual es mostrado en el algoritmo 5, todos los puntos considerados como “daño” quedan contenidos en Dmg_Obj . Como nuestra propuesta es de *inspección activa*, necesitamos saber si la región con daño continúa más allá de lo encontrado hasta el momento, es decir, si se requiere seguir inspeccionando el objeto de prueba. Para ello obtenemos desde Dmg_Obj sus valores límites de volumen, y así compararlos con los valores límites de T_Obj . De esta forma, cuando estos límites se encuentran cercanos, se efectúa un nuevo ciclo de rotación de la plataforma giratoria, hasta completar totalmente el daño.

Algoritmo 5: Función Detección

Entrada: T_Obj %Reconstrucción 3D del objeto de prueba,
 $Align_SObj$ %Reconstrucción 3D alineada del objeto base.

Salida: Dmg_Obj %Reconstrucción 3D del daño encontrado en el objeto de prueba.

```

1 Función  $Dmg\_Obj \leftarrow Detección(Align\_SObj, T\_Obj)$ 
2    $i = 1$  a  $length(T\_Obj)$   $Ind\_1 \leftarrow KNN(T\_Obj(i), Align\_SObj)$ ;
3    $j = 1$  a  $length(Align\_SObj)$   $Ind\_2 \leftarrow KNN(Align\_SObj(j), T\_Obj)$ ;
4    $maxDist\_1 \leftarrow$  Máximo valor de  $Ind\_1$ ;
5    $maxDist\_2 \leftarrow$  Máximo valor de  $Ind\_2$ ;
6   si  $maxDist\_1 < maxDist\_2$  entonces
7     para  $k = maxDist\_2$  a  $minDist$  hacer
8        $Dmg\_Obj(k) \leftarrow$  Filtrar  $Ind\_2$  con respecto a  $k$  ;
9     fin
10  en otro caso
11    para  $k = maxDist\_1$  a  $minDist$  hacer
12       $Dmg\_Obj(k) \leftarrow$  Filtrar  $Ind\_1$  con respecto a  $k$  ;
13    fin
14  fin
15   $Dmg\_Obj \leftarrow$  Filtrar datos anormales y repetidos;
16 fin

```

3.8. Cálculo de Volumen

Una vez terminada la etapa de detección de daño y, por ende, determinada toda la región de daño en el objeto, comenzamos con el cálculo de volumen. Para esto, necesitamos transformar a $Dmg.Obj$, que actualmente es una reconstrucción tridimensional creada a través de una nube de puntos a una reconstrucción con una superficie más cerrada y definida. Con esa finalidad en mente, utilizamos *AlphaShape* [Gardiner *et al.*, 2018; Liao *et al.*, 2021]. Éste un concepto geométrico, el cual es una generalización del casco convexo y un subgrafo de la triangulación de Delaunay, que permite la composición de polígonos dado un conjunto de puntos y así generar un volumen delimitador que envuelve a la nube de puntos. Su ejecución está sustentada bajo la siguiente lógica; Dado un conjunto de puntos finito, se puede derivar una familia de formas mediante la triangulación de Delaunay con dichos puntos; un parámetro real " α ", el cual controla el nivel de detalle deseado. Dicho valor α está definido conceptualmente de la siguiente forma:

Para cada valor α , definimos un disco generalizado de radio $1/\alpha$ tal que:

- Si $\alpha > 0$, es un disco cerrado ordinario de radio $1/\alpha$
- Si $\alpha = 0$, es un semiplano cerrado
- Si $\alpha < 0$, es el complemento de un disco cerrado de radio $-1/\alpha$

Luego, dado un conjunto de puntos y un valor específico para α , construimos el gráfico de forma α bajo la siguiente directriz:

Para cada punto p_i en nuestro conjunto de puntos, creamos un vértice v_i . Creamos una arista entre dos vértices v_i y v_j siempre que exista un disco generalizado de radio $1/\alpha$ que contenga todo el conjunto de puntos y que tenga la propiedad de que p_i y p_j se encuentran en su límite.

Todo esto puede ser explicado de manera más simple a través de la figura 3.5. En 3.5a se aprecia una reconstrucción de una región de objeto Pato mediante una nube de puntos, la cual no presenta una superficie definida. Sin embargo, al utilizar *AlphaShape* en dicha reconstrucción 3D, se crea una superficie creada por una serie de polígonos que encierran al conjunto de puntos que conforman al modelo 3D, como se logra apreciar en la figura 3.5b. Esto nos permite poder realizar métricas más realistas sobre la reconstrucción del objeto.

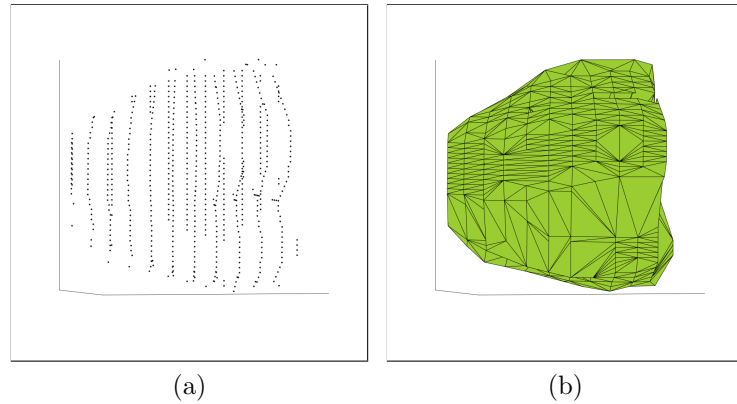


Figura 3.5: Representación de *AlphaShape*. (a) Región del objeto Pato reconstruida mediante nube de puntos, (b) Región del objeto Pato creada utilizando *AlphaShape* sobre la nube de puntos.

Una vez recreado la superficie del objeto con *AlphaShape*, se realiza el cálculo de volumen de la región de daño. Para obtener dicho valor primero es necesario realizar el cálculo de volumen para cada polígono, de la siguiente forma:

$$V_p = Largo \cdot Ancho \cdot Alto \quad (3.18)$$

Luego de calcular el volumen para cada polígono V_p que compone al modelo de la región dañada, sólo se debe obtener el volumen total de daño Vol , el cual sería la sumatoria de los n polígonos generados por el *AlphaShape*:

$$Vol = \sum_1^n V_p \quad (3.19)$$

El procedimiento anterior, se describe de mejor manera en el algoritmo 6.

Algoritmo 6: Función Calc_Vol

Entrada: Dmg_Obj %Reconstrucción 3D del daño encontrado en el objeto de prueba.

Salida: Vol %Volumen de daño.

```
1 Función  $Vol \leftarrow Calc\_Vol(Dmg\_Obj)$ 
2    $AS\_Dmg \leftarrow$  Realizar AlphaShape a  $Dmg\_Obj$ ;
3    $n\_p \leftarrow$  Número de polígonos creados en  $AS\_Dmg$ ;
4   para  $n = 1$  a  $n\_p$  hacer
5      $V_p(n) \leftarrow$  Ancho de  $AS\_Dmg(p) \times$  Alto de  $AS\_Dmg(p) \times$  Largo de
6      $AS\_Dmg(p)$ ;
7   fin
8    $Vol \leftarrow \sum_1^p V_p$ ;
9 fin
```

Capítulo 4

Evaluación

Para validar el desempeño de la metodología propuesta, hemos implementado un sistema que considera todo lo definido anteriormente y el diseño mostrado en la Figura 3.1, el cual nos permite realizar pruebas utilizando diferentes objetos, que pueden -o no- poseer algún daño ¹.

4.1. Diseño del Sistema

Los componentes utilizados en el sistema detector de daño son los siguientes: un monitor marca ViewSonic, un computador Dell Octiplex 7050 con procesador Intel Core i7-7000 de 3.60 GHz, un sistema de adquisición de imágenes y un controlador de motor-líneas láser (ver Figura 4.1).

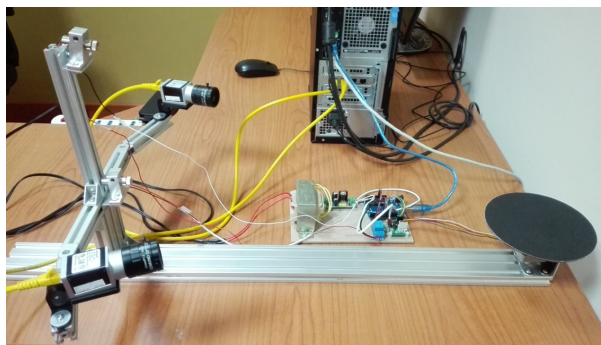


Figura 4.1: Sistema detector de daño.

¹Los códigos y una base de datos restringida con algunos ejemplos de prueba (off-line) están disponibles en el siguiente enlace de descarga: <https://drive.google.com/drive/folders/1SAV1TA7-9l5vU7msq-TC6R2xHfV86cHa?usp=sharing>

El computador tiene incorporado una tarjeta de adquisición de imágenes, NI PCIe-8233 Quad Port Gigabit Ethernet Interface Device, para controlar las capturas simultaneas obtenidas de las cámaras a través de conectores Ethernet 8-pin RJ-45 Jack.

El sistema de inspección está formada por una plataforma giratoria que utiliza un motor paso a paso RS 440-420 de 1.8° , 2 diodos láser ajustables de 5mW y 2 cámaras Basler acA1280-60gc con lentes Edmund Optics CFFL F1.3 f8.5mm. El controlador de movimiento está constituido por una placa Arduino Uno, un módulo controlador de motor, un módulo relé de 1 canal, un regulador de voltaje y una fuente de poder.

4.2. Pruebas de Desempeño

Utilizamos 5 objetos, mostrados en la Figura 4.2, para realizar una serie de pruebas en tres diferentes escenarios de inspección: *i*) inspección a objeto sin daño, *ii*) inspección a objeto con daño aditivo e *iii*) inspección a objeto con daño sustractivo. Además, cada prueba fue repetida 3 veces utilizando distintas posiciones iniciales del objeto con la finalidad de observar la cantidad de rotaciones dadas por el sistema para detectar y calcular el volumen del daño (en caso de existir).

Para corroborar el cálculo de volumen obtenido, todos los daños presentes en los objetos han sido creados intencionalmente y su volumen es conocido.

4.2.1. Inspección a objetos sin daño

El requisito para realizar este tipo de inspección es simple: utilizar el objeto sin alterar, es decir, los objetos que se muestran en la Figura 4.2. Hemos realizado 5 pruebas de este tipo (una para cada objeto) y fueron denominadas ND (No Damage).

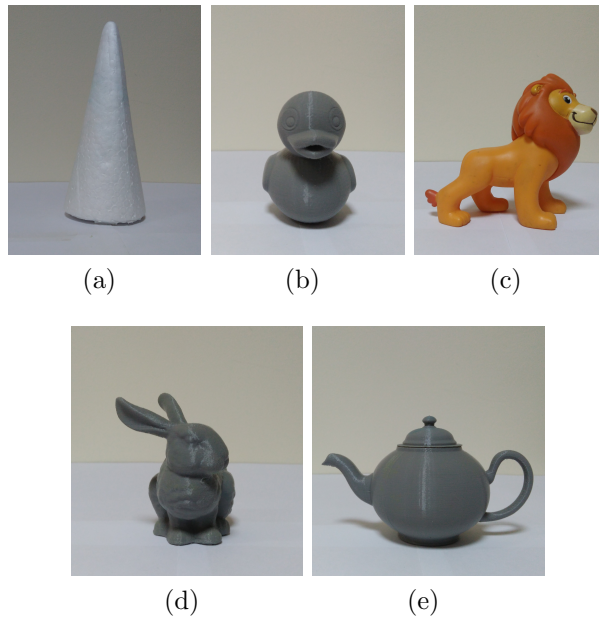


Figura 4.2: Objetos de prueba. (a) Cono, (b) Pato, (c) León, (d) Conejo Stanford, y (e) Tetera.

4.2.2. Inspección a objetos con daño aditivo

Para llevar a cabo este tipo de pruebas, incorporamos volúmenes conocidos a los objetos de prueba, mediante el uso de plastilina. Para comprobar que el volumen calculado con nuestro sistema coincida con el volumen añadido, hemos construido tres cubos de plastilina, para adicionar a los 5 objetos de manera uniforme en cualquier lugar de éstos. Los cubos son 10 [mm], 15 [mm] y 20 [mm] de lado, y por lo tanto, con volúmenes de 1000 [mm³], 3375 [mm³] y 8000 [mm³], respectivamente (ver Figura 4.3).

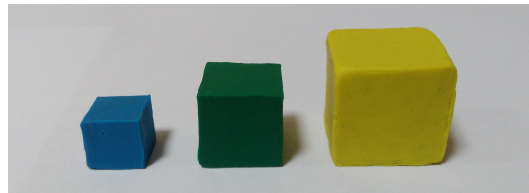


Figura 4.3: Cubos de plastilina de 1000 [mm³], 3375 [mm³] y 8000 [mm³].

Hemos realizado i inspecciones con daño aditivo para cada objeto, denominadas AD, es decir, AD i , donde $i = 1, 2, 3$. Dependiendo del valor de i , utilizamos un cubo de plastilina para adicionar al objeto; Si $i = 1$ utilizamos el cubo de 1000 [mm³], si

$i = 2$ utilizamos el cubo de 3375 [mm³] y si $i = 3$ utilizamos el cubo de 8000 [mm³].

Para cada prueba de inspección, hemos adicionado sólo un cubo de plastilina al objeto, como se puede apreciar en las Figuras 4.4a, 4.4b y 4.4c que muestran al objeto utilizado para las pruebas Conejo Stanford AD1, Tetera AD2 y Cono AD3, respectivamente.

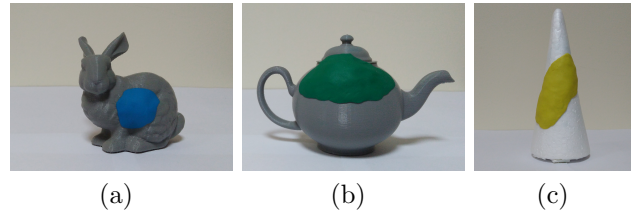


Figura 4.4: Objetos con daño aditivo. (a) Conejo Stanford AD1, (b) Tetera AD2, y (c) Cono AD3.

4.2.3. Inspección a objetos con daño sustractivo

Para este tipo de inspección, hemos duplicado mediante impresión 3D algunos de los objetos (Pato, Conejo Stanford y Tetera) con pequeñas diferencias, las cuales fueron eliminadas para considerarlo como daño. En el caso del cono de poliestireno expandido, lo alteramos manualmente para realizar la inspección.

Hemos realizado i inspecciones con daño sustractivo para cada objeto, denominadas SD, es decir, SD_i , donde $i = 1, 2, 3$. En la Figura 4.5a, 4.5b y 4.5c se muestran los objetos utilizados para realizar las pruebas Pato SD1, Conejo Stanford SD2 y Cono SD3 respectivamente.

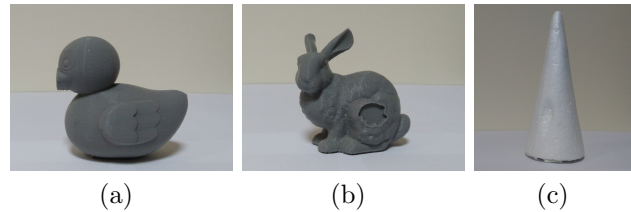


Figura 4.5: Objetos con daño sustractivo. (a) Pato SD1, (b) Conejo Stanford SD2, y (c) Cono SD3.

4.3. Análisis de Resultados

Hemos realizado un total de 84 inspecciones, 15 a objeto sin daño, 45 a objetos con daño aditivo y 24 a objetos con daño sustractivo. Podemos ver en la Figuras 4.6 y 4.7, algunas vistas de las reconstrucciones 3D de los daños, obtenidas desde las pruebas realizadas a los objetos que se muestran en la Figuras 4.4 y 4.5 respectivamente.

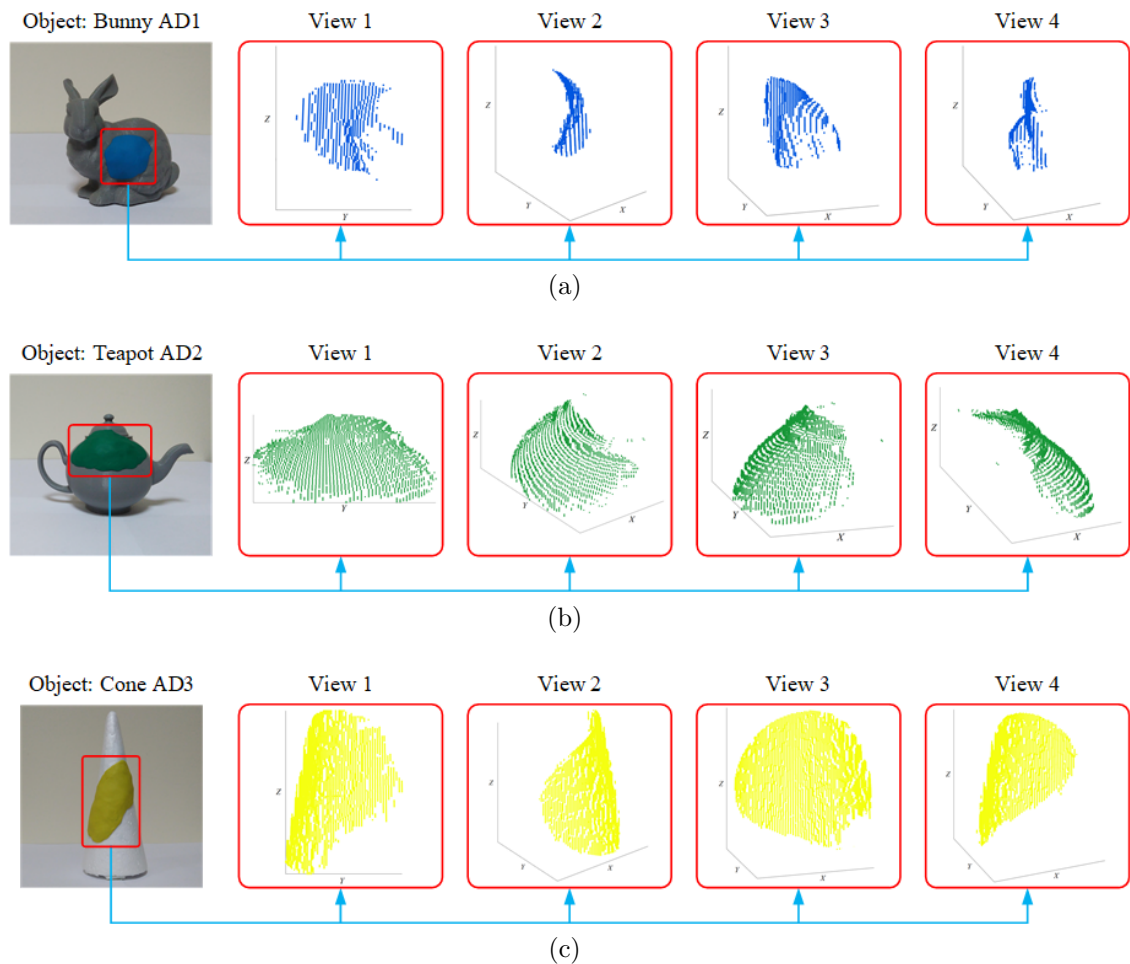


Figura 4.6: Reconstrucción 3D de daño aditivo para las pruebas: (a) Conejo Stanford AD1, (b) Tetera AD2, y (c) Cono AD3.

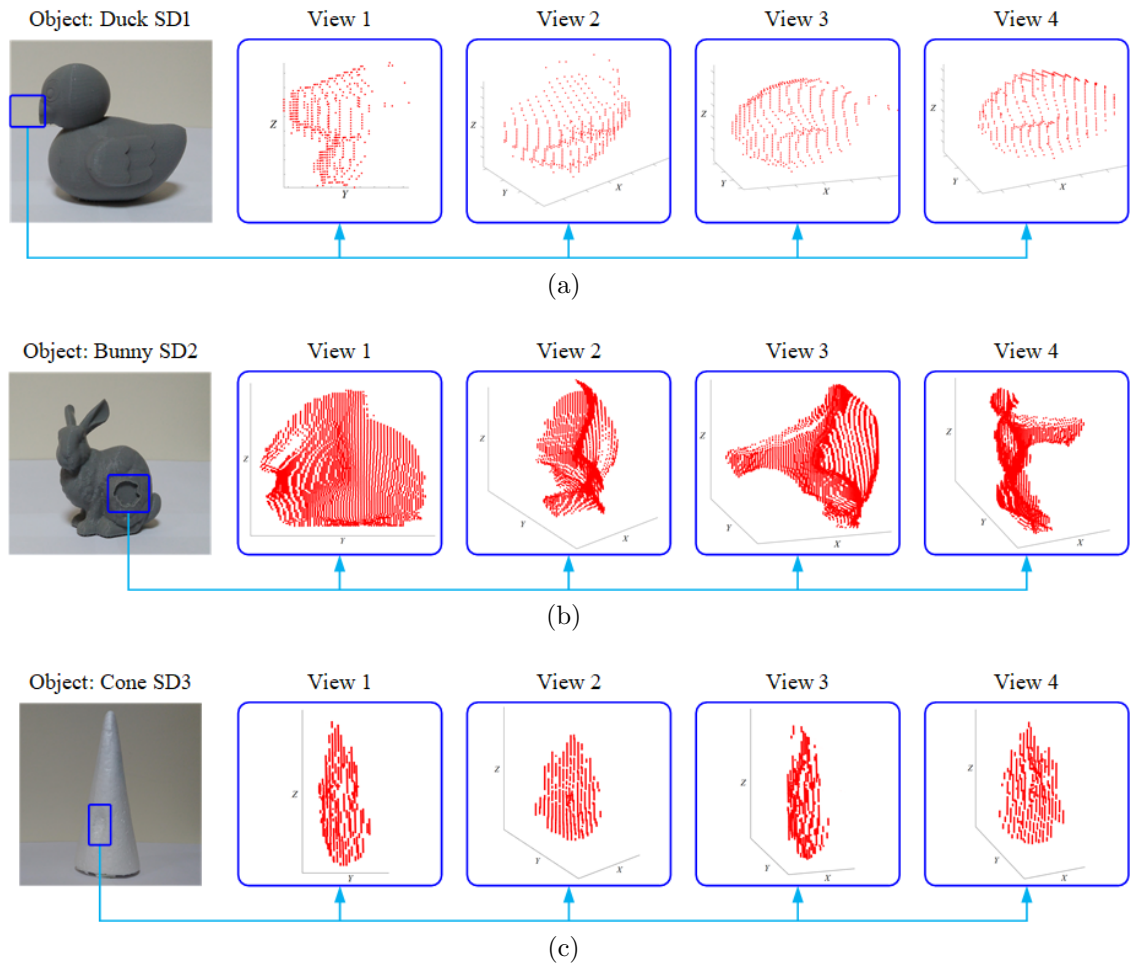


Figura 4.7: Reconstrucción 3D de daño sustractivo para las pruebas: (a) Duck SD1, (b) Stanford Bunny SD2, y (c) Cone SD3.

Para evaluar el desempeño de la metodología propuesta, hemos usado el *Error porcentual* ($error$ [%]) y *Error porcentual absoluto medio* ($MAPE$). Además, contabilizamos las rotaciones necesarias para lograr la detección de un eventual daño. Estos resultados se muestran en la Tabla 4.1.

El $error$ [%] se calcula usando la siguiente ecuación (4.1):

$$error[\%] = \frac{|Volr - Vol|}{Volr} \cdot 100, \quad (4.1)$$

donde, $Volr$ es el volumen real del daño, y Vol es el volumen estimado del daño calculado con nuestro método.

Tabla 4.1: Pruebas de rendimiento para el sistema de detección de daño.

Objeto	Prueba	Número de rotaciones para completar la prueba			Volumen de daño real, [mm ³]	Volumen de daño estimado, [mm ³]	error[%]
		Obj 1	Obj 2	Obj 3	Volr	Vol	
Cono	ND	6	6	6	0	0	0,00
	AD1	5	5	6	1000	959	4.10
	AD2	3	3	7	3375	3173	5.99
	AD3	7	9	8	8000	7565	5.44
	SD1	11	11	11	4570	4350	4.81
	SD2	6	4	5	2845	2640	7.21
	SD3	1	1	4	3266	3055	6.46
Pato	ND	6	6	6	0	0	0,00
	AD1	4	2	2	1000	873	12.70
	AD2	4	5	3	3375	3166	6.19
	AD3	7	9	9	8000	7781	2.74
	SD1	1	3	5	9744	9558	1.91
León	ND	6	6	6	0	0	0,00
	AD1	1	3	5	1000	890	11.00
	AD2	1	1	1	3375	3356	0.56
	AD3	1	6	7	8000	7981	0.24
Conejo Stanford	ND	6	6	6	0	0	0,00
	AD1	1	2	7	1000	950	5.00
	AD2	1	8	4	3375	3356	0.56
	AD3	1	8	10	8000	7981	0.24
	SD1	1	1	1	2210	2200	0.45
	SD2	1	4	4	3450	3420	0.87
	SD3	1	2	7	4200	4188	0.29
Tetera	ND	6	6	6	0	0	0,00
	AD1	1	11	11	1000	995	0.50
	AD2	1	5	7	3375	3369	0.18
	AD3	1	4	4	8000	7989	0.14
	SD1	1	11	11	9527	9500	0.28
<i>MAPE</i> =							2.78

Para determinar el error global de nuestra propuesta en las pruebas realizadas, debemos calcular el *Error porcentual absoluto medio (MAPE)* usando la ecuación (4.2):

$$MAPE = \frac{1}{n} \sum_{i=1}^n error[\%]_i \quad (4.2)$$

donde, n es la cantidad de pruebas realizadas.

De los datos que se muestran en la Tabla 4.1 podemos indicar que se realizaron 21 pruebas tanto para el objeto Cono como para el objeto Conejo Stanford; 3 fueron ND (No Damage), 9 fueron AD (Additive Damage) y 9 fueron SD (Subtractive Damage). Además, se realizaron 15 ensayos para los objetos Pato y Tetera, 3 de los cuales fueron ND, 9 AD y 3 SD. Para el objeto León solo se realizaron 12 pruebas, 3 ND y 9 AD. Vemos que los errores porcentuales obtenidos para la prueba Pato AD1

obtuvieron el valor más alto con un 12,70%, seguido de la prueba León AD1, que obtuvo un error porcentual igual al 11%. Sin considerar los resultados obtenidos en las pruebas de ND de cada objeto, el mejor valor lo obtuvo la prueba Tetera SD3 con 0.14% que corresponde a 11 [mm³] de diferencia entre el volumen real Vol_r y el volumen estimado Vol . En relación a las rotaciones realizadas para detectar y cuantificar daños en esta propuesta de inspección activa, van de 1 a 11 rotaciones, las cuales fueron necesarias para completar el proceso de inspección de objetos. Cabe señalar que en todas las pruebas de ND se realizaron un total de 6 rotaciones para completar la inspección y asegurar que no hubo daños.

Finalmente, el sistema de inspección y cálculo de daño, arrojó un valor de $MAPE = 2,78\%$, por lo cual podemos decir que nuestra propuesta permite realizar una inspección de objetos muy eficiente y con volúmenes de daño calculado cercanos a los valores reales.

Además, podemos ver en la Tabla 4.1 que para las pruebas con daño existente en el objeto de prueba, el sistema es capaz de ejecutar una cantidad de rotaciones adecuadas para poder reconstruir y calcular el volumen del daño detectado, reduciendo los pasos para completar la inspección.

La principal restricción de nuestro método es que la rotación solo se realiza para el eje Y y hemos utilizado solo dos cámaras, que permiten adquirir la deformación de las líneas láser. Hay áreas del objeto de prueba que no se pueden reconstruir y, en consecuencia, no se puede detectar ningún daño (si lo hay). Esto puede ocurrir en los extremos superior e inferior del objeto y en áreas intrincadas, donde la línea láser no está bien definida o simplemente porque la línea láser no se cruza con la superficie del objeto. Esto se puede evitar realizando dos inspecciones adicionales del objeto; uno girando el objeto 90° y otro con el objeto girado a -90°, ambos para el eje X . Otra solución a este problema podría ser incorporar más líneas láser y cámaras, lo que permitiría inspeccionar más zonas de difícil acceso.

Capítulo 5

Conclusiones

En este trabajo implementamos una metodología de detección de daño presente en objetos utilizando un enfoque basado en la búsqueda activa. El beneficio que presenta radica en su eficacia al inspeccionar un objeto de prueba, ya que lo realiza por secciones (cantidad definida de capturas obtenidas por cada rotación dada por la plataforma giratoria) hasta que encuentre un área incongruente en comparación a un modelo del objeto sin daños, almacenado en una base de datos.

Si bien existen trabajos en la inspección de objetos así como también en la detección de daño, no encontramos, durante nuestra búsqueda, literatura sobre inspección activa de objetos y cuantificación del daño detectado, los cuales son el centro de este trabajo, para poder realizar comparaciones con otros métodos. Esto hace que nuestro trabajo sea bastante novedoso en la inspección de objetos.

Para comprobar el desempeño del sistema de inspección, hemos realizado 84 pruebas, en tres escenarios: *i*) inspección a objeto sin daño, *ii*) inspección a objeto con daño aditivo e *iii*) inspección a objeto con daño sustractivo. Los resultados que hemos obtenidos son auspiciosos e indican que el sistema que aquí proponemos, posee un *Error porcentual absoluto medio MAPE* = 2,78%, confirmando que el sistema presenta un análisis de inspección aceptable.

El mayor *Error porcentual (error [%])* lo hemos obtenido en los objetos con daño aditivo de tamaño 1000 [mm³], y esto puede ser causado por la zona donde fue incorporado dicho daño, pudiendo ser una zona intrincada, donde la línea láser no queda bien definida y/o las posición de las cámaras impiden la visualización total de la línea láser.

Cabe destacar que este sistema es un primer acercamiento a esta metodología, y

viendo los resultados obtenidos, pensamos que algunos aspectos relevantes no fueron considerados, tales como el tiempo de inspección, secciones del objeto no analizadas y por lo tanto no reconstruidos debido al número limitado de cámaras, entre otros aspectos. Como trabajo futuro, proponemos mejorar la estructura de adquisición de imágenes de manera que nos permita capturar imágenes que incluyan regiones que actualmente no fueron totalmente observadas y, por consecuencia, no reconstruidas tridimensionalmente tales como la parte inferior y superior del objeto. En otras palabras, aumentar el número de cámaras y dar mayor libertad de posicionamiento a éstas, para mayor cobertura de inspección.

Referencias

- Cirujeda, P., Dicente Cid, Y., Mateo, X., y Binefa, X. (2015). A 3D scene registration method via covariance descriptors and an evolutionary stable strategy game theory solver. *Int. J. Comput. Vision*, 115:306–329. 23
- Cock, J. (2012). The triangulation method applied in a laser scanner, for three-dimensional objects. *EAFIT University Magazine [in Spanish]*, 36:25–31. 14
- Czimmermann, T., Ciuti, G., Milazzo, M., Chiurazzi, M., Roccella, S., Oddo, C., y Dario, P. (2020). Visual-based defect detection and classification approaches for industrial applications -A survey. *Sensors*, 20:1459. 1, 5
- Dawda, A. y Nguyen, M. (2020). Defects detection in highly specular surface using a combination of stereo and laser reconstruction. pp. 1–6. 6
- Dawda, A., Nguyen, M., y Klette, R. (2019). Accurate 3d measurement of highly specular surface using laser and stereo reconstruction. pp. 1–6. 6
- Drouin, M.-A. y Beraldin, J.-A. (2020). *Active Triangulation 3D Imaging Systems for Industrial Inspection*, pp. 109–165. 14
- Gardiner, J., Behnsen, J., y Brassey, C. (2018). Alpha shapes: determining 3D shape complexity across morphologically diverse structures. *BMC Evolutionary Biology*, 18. 27
- Gonzalez, R. y Woods, R. (2018). *Digital Image Processing*. Pearson. 16
- Guan, J., Yang, X., Ding, L., Cheng, X., Lee, V., y Jin, C. (2021). Automated pixel-level pavement distress detection based on stereo vision and deep learning. *Automation in Construction*, 129:103788. 8

- Güldür Erkal, B. y Hajjar, J. (2020). Using extracted member properties for laser-based surface damage detection and quantification. *Structural Control and Health Monitoring*, 27. 6
- Gupta, M., Khan, M., Butola, R., y Singari, R. (2021). Advances in applications of non-destructive testing (NDT): A review. *Advances in Materials and Processing Technologies*, pp. 1–22. 14
- Isa, M. y Lazoglu, I. (2017). Design and analysis of a 3d laser scanner. *Measurement*, 111. 2
- Kus, A. (2009). Implementation of 3D optical scanning technology for automotive applications. *Sensors (Basel, Switzerland)*, 9:1967–1979. 14
- Kwon, H., Na, M., y Song, J.-B. (2019). Rescan strategy for time efficient view and path planning in automated inspection system. *International Journal of Precision Engineering and Manufacturing*, 20:1747–1756. 6
- Liao, Z., Liu, J., Shi, G., y Meng, J. (2021). Grid partition variable step alpha shapes algorithm. *Mathematical Problems in Engineering*, 2021:1–8. 27
- Liu, C., Zhou, L., Wang, W., y Zhao, X. (2021). Concrete surface damage volume measurement based on three-dimensional reconstruction by smartphones. *IEEE Sensors Journal*, 21(10):11349–11360. 8
- Liu, Y., Wang, Y., Cai, X., y Hu, X. (2020). The detection effect of pavement 3D texture morphology using improved binocular reconstruction algorithm with laser line constraint. *Measurement*, 157:107638. 2
- Long, L. y Dongri, S. (2019). *Review of Camera Calibration Algorithms*, pp. 723–732. 15
- Mayorca, D. y López Rubio, A. (2018). Reconstruction of 3D objects with axial symmetry from laser triangulation. *UNIMAR Magazine [in Spanish]*, 35:239–253. 14
- Mishra, V. K., Kumar, S., y Shukla, N. (2017). Image acquisition and techniques to perform image acquisition. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology*, 9. 14

- Mitra, S., Pal, S. K., y Mitra, P. (2002). Data mining in soft computing framework: a survey. *IEEE Transactions on Neural Networks*, 13:3–14. 10
- Molleda, J., Usamentiaga, R., Garcia, F. D., Bulnes, F., y Ema, L. (2010). Shape measurement of steel strips using a laser-based three-dimensional reconstruction technique. *IEEE Transactions on Industry Applications - IEEE TRANS IND APPL*, 47:1–8. 5
- Molleda, J., Usamentiaga, R., Garcia, F. D., Bulnes, F., Espina, A., Dieye, B., y Smith, L. (2013). An improved 3D imaging system for dimensional quality inspection of rolled products in the metal industry. *Computers in Industry*, 64:1186–1200. 5
- Nikonova, G., Nikonov, A., Zakirova, Y., y Makarochkin, V. (2021). Automated system for visual non-destructive testing. pp. 1–5. 2
- Phillips, J. M. (2021). Distances and nearest neighbors. En *Mathematical Foundations for Data Analysis*, pp. 59–94. Springer. 25
- Riffo, V. y Mery, D. (2016). Automated detection of threat objects using adapted implicit shape model. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46:472–482. 1
- See, J., Drury, C., Speed, A., Williams, A., y Khalandi, N. (2017). The role of visual inspection in the 21 st century. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 61:262–266. 1
- Shim, S., Kim, J., Cho, G.-C., y Lee, S.-W. (2022). Stereo-vision-based 3d concrete crack detection using adversarial learning with balanced ensemble discriminator networks. *Structural Health Monitoring*, 22:147592172210978. 9
- Tian, Z., Jianhua, L., Shaoli, L., Chengtong, T., y Peng, J. (2016). A 3D reconstruction method for pipeline inspection based on multi-vision. *Measurement*, 98. 2
- Tyagi, V. (2018). *Understanding Digital Image Processing*. 16
- Vundekode, N., Kalapatapu, P., y Pasupuleti, V. (2021). *A Study on Vision Based Method for Damage Detection in Structures*, pp. 96–105. 2

- Wang, H., Dntsch, I., y Gediga, G. (2003). Nearest neighbours without k: A classification formalism based on probability. 25
- Wang, Y., Deng, N., y Xin, B. (2020). Investigation of 3d surface profile reconstruction technology for automatic evaluation of fabric smoothness appearance. *Measurement*, 166:108264. 2
- Zhang, B., Guo, N., Huang, J., Gu, B., y Zhou, J. (2020). Computer vision estimation of the volume and weight of apples by using 3d reconstruction and noncontact measuring methods. *Journal of Sensors*, 2020:1–12. 7
- Zhang, Z. (2000). A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22:1330–1334. 15, 20
- Zidane, F., Lanteri, J., Migliaccio, C., y Marot, J. (2021). System measurement optimized for damages detection in fruit. En *2021 IEEE Conference on Antenna Measurements Applications (CAMA)*, pp. 550–554. 7