

Trabajo Fin de Grado  
Grado en Ingeniería Electrónica, Robótica y  
Mecatrónica

Desarrollo de un sistema de estantería inteligente  
mediante visión por computador

Autor: Álvaro Muñoz Velasco

Tutor: Carlos Vivas Venegas

Dpto. Ingeniería de Sistemas y Automática  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2020





Trabajo Fin de Grado  
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

# **Desarrollo de un sistema de estantería inteligente mediante visión por computador**

Autor:

Álvaro Muñoz Velasco

Tutor:

Carlos Vivas Venegas

Profesor titular

Dpto. de Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2020



Trabajo Fin de Grado: Desarrollo de un sistema de estantería inteligente mediante visión por computador

Autor: Álvaro Muñoz Velasco

Tutor: Carlos Vivas Venegas

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Sevilla, 2020



# Agradecimientos

---

Este Trabajo Fin de Grado no podría haber llegado a su fin de no ser por el apoyo de aquellas personas que me han acompañado en el camino. A mi familia, por su cariño y comprensión durante los largos periodos de confinamiento iniciados con 4 años de antelación al COVID-19. Ellos siempre han demostrado su gran apoyo ante cualquier idea y decisión propia.

También quiero agradecerlo a Belén, por ser el pilar fundamental en todos los entornos de mi vida y por su constante motivación para afrontar todas mis metas. Y a todos mis amigos repartidos por cada rincón del mundo, que me animan a seguir adelante.

Gracias a AIRBUS, por hacer posible la realización de este proyecto, y a todas las personas de la empresa que han dedicado su tiempo y esfuerzo, en especial a mis tutores David Vega y Paco Sierra por su confianza y guía, y a mis compañeros de departamento.

Por último, pero no menos importante, quiero agradecerle a mi tutor académico Carlos Vivas su ayuda y disposición durante este proceso. Una persona que aprecio y admiro, y que considero un gran referente a seguir.

*Álvaro Muñoz Velasco*

*Sevilla, 2020*



# Resumen

---

Este proyecto aborda un problema real de localización del inventario de la empresa AIRBUS con el objetivo de reducir tiempos de fabricación en el sector aeroespacial. Así pues, se hace uso de la visión artificial para diseñar un programa capaz de detectar e identificar las piezas que se encuentran almacenadas en cajas situadas en estanterías, para tener una digitalización de los datos y poder actuar en consecuencia.

Con el fin de la implantación de esta y otras tecnologías en auge de la Industria 4.0, se precisa de un rediseño del entorno físico, llevado a cabo en este documento. Por ello, se dota a las estanterías de visión artificial mediante cámaras, y se idean nuevas cajas de almacenaje abiertas y silueteadas que permitan tener un orden y control de las piezas en su interior.

Las metodologías aplicadas incluyen el procesamiento de imágenes para determinar la presencia o no de las piezas en las cajas, y la extracción de características locales para identificar si las piezas encontradas en cada hueco se corresponden con las que se buscan.

Este Trabajo Fin de Grado es la base para un posterior desarrollo que permita la implantación completa de esta solución, junto a las demás herramientas y programas existentes.



# Abstract

---

This work addresses a real problem about inventory tracking in AIRBUS company, to reduce manufacturing times in the aerospace sector. To deal with it, artificial vision is used to design specific software that allows to detect and identify the parts and tools that are stored in boxes located on shelves, so the data is digitized in real time and is able to act accordingly.

To implement this and other new technologies of 4.0 Industry, a physical redesign is tackled in this work. Therefore, the shelves are equipped with artificial vision using cameras, and new open and silhouetted boxes are designed that allow to have a better control of the pieces inside it.

Applied methodologies include image processing to determine the presence or not of parts in boxes, and the extraction of keypoints to identify whether the parts found in each gap correspond to those sought.

This research is the base for further development that allows the complete implementation of this solution, together with the other existing tools and programs.



# Índice

<b>Agradecimientos</b>	<b>i</b>
<b>Resumen</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Índice</b>	<b>vii</b>
<b>Índice de Tablas</b>	<b>ix</b>
<b>Índice de Figuras</b>	<b>xi</b>
<b>1 Introducción</b>	<b>1</b>
1.1 <i>Justificación</i>	1
1.2 <i>Objetivo</i>	1
1.3 <i>Estructura del documento</i>	2
<b>2 Localización de Materiales por Visión Artificial</b>	<b>5</b>
2.1 <i>Antecedentes de la localización de materiales</i>	5
2.2 <i>Visión artificial para la detección de objetos</i>	6
2.2.1 <i>Similitudes entre la visión artificial y la visión humana</i>	7
2.2.2 <i>Imágenes digitales</i>	8
2.2.3 <i>Histograma</i>	8
2.3 <i>Estado del arte</i>	9
2.3.1 <i>Retos de la Visión Artificial</i>	9
<b>3 Descripción formal del problema</b>	<b>11</b>
3.1 <i>Elección del problema</i>	11
3.2 <i>Definición del problema</i>	15
3.3 <i>Acotación del problema</i>	17
<b>4 Descripción de la metodología</b>	<b>19</b>
4.1 <i>Elección de la tecnología</i>	19
4.1.1 <i>RFID</i>	19
4.1.2 <i>Visión artificial</i>	20
4.2 <i>Métodos de visión artificial para la localización de materiales</i>	21
4.2.1 <i>Procesamiento de imágenes</i>	21
4.2.2 <i>Extracción de características locales</i>	25
4.2.3 <i>Redes neuronales</i>	28
4.3 <i>Entorno de programación</i>	29
4.3.1 <i>Python</i>	29
4.3.2 <i>OpenCV</i>	30
<b>5 Desarrollo de la aplicación</b>	<b>31</b>
5.1 <i>Entorno físico</i>	31
5.1.1 <i>Estantería</i>	31
5.1.2 <i>Cajas</i>	33
5.1.3 <i>Cámaras</i>	33
5.2 <i>Estructura de los algoritmos</i>	33
5.3 <i>Interfaz gráfica: InterfazGrafica.py</i>	34
5.3.1 <i>Diseño de la aplicación</i>	34

---

5.3.2	Algoritmo de la aplicación	36
5.4	<i>Funciones de visión: Funciones.py</i>	36
5.5	<i>Conexión de la interfaz con la funcionalidad: Principal.py</i>	42
<b>6</b>	<b>Resultados experimentales</b>	<b>45</b>
6.1	<i>Interfaz gráfica y funcionamiento general</i>	45
6.2	<i>Resultados de los métodos SIFT y SURF</i>	47
<b>7</b>	<b>Conclusiones y líneas futuras</b>	<b>53</b>
	<b>Referencias</b>	<b>55</b>

# ÍNDICE DE TABLAS

---

Tabla 5.1. Significado de los colores de las cajas en la aplicación	35
Tabla 5.2 Excepciones y su significado.	43



# ÍNDICE DE FIGURAS

---

Figura 2.1. Quipus Incas.	5
Figura 2.2. Partes del ojo humano.	7
Figura 2.3. Composición de filtros para separación de fotones según su longitud de onda.	8
Figura 2.4. Comparación de histogramas para imágenes con diferentes contrastes.	9
Figura 3.1. Diseño propio de sistema identificador de materiales.	11
Figura 3.2. Sistema de reconocimiento facial en aeropuertos.	12
Figura 3.3. Diseño propio asistente virtual con reconocimiento facial.	12
Figura 3.4. Comprobación del uso de EPIs con visión artificial.	13
Figura 3.5. Control de flujo de personas para el rediseño de las calles.	13
Figura 3.6. Realidad aumentada con visión artificial para identificación de elementos.	14
Figura 3.7. Inspección de alta precisión con visión artificial.	14
Figura 3.8. Lector de indicadores numéricos analógicos por visión.	15
Figura 3.9 Recorrido de los paquetes de la empresa.	16
Figura 4.1. Etiqueta RFID.	19
Figura 4.2. Lector RFID.	20
Figura 4.3. Resultados de distintos tipos de binarización.	23
Figura 4.4. Resultado de la binarización Otsu.	23
Figura 4.5. Resultado de la operación de búsqueda de contornos.	24
Figura 4.6. Comparativa de operador Canny y Sobel.	24
Figura 4.7. Posibles rasgos característicos de las imágenes.	25
Figura 4.8. Rasgos simples de las imágenes.	26
Figura 4.9. Varianza del escalado en descriptores Harris y Shi-Tomasi.	26
Figura 4.10. Correspondencia de puntos característicos con el método SIFT.	27
Figura 4.11. Comparación repositorios OpenCV para diferentes lenguajes de programación en GitHub.	30
Figura 5.1. Vista 1, diseño propio de la estantería para visión artificial.	32
Figura 5.2. Vista 2, diseño propio de la estantería para visión artificial.	32
Figura 5.3 Vista 2, diseño propio de las cajas para visión artificial.	33
Figura 5.4 Vista 1, diseño propio de las cajas para visión artificial.	33
Figura 5.5. Estructura de los algoritmos.	34

---

Figura 5.6. Diseño aplicación, pestaña etiquetas.	35
Figura 5.7. Diseño aplicación, pestaña cajas.	36
Figura 5.8. Foto de la caja con puntos de referencia.	37
Figura 5.9. Caja discretizada por color rojo.	37
Figura 5.10. Caja recortada por puntos de referencia.	38
Figura 5.11. Recorte de la caja para analizar código QR.	38
Figura 5.12. Proceso de búsqueda de caja de referencia en base de datos.	39
Figura 5.13. Mascara binarizada de la caja de referencia.	39
Figura 5.14. Resultados búsqueda de materiales por conteo de píxeles rojos.	40
Figura 5.15. Resultado de identificación de materiales por extracción de características locales.	41
Figura 6.1. Interfaz final: vista etiquetas de caja con algunas piezas.	45
Figura 6.2. Interfaz final: vista imagen real de caja con algunas piezas.	45
Figura 6.3. Interfaz final: vista imagen real de caja con todas sus piezas.	46
Figura 6.4. Interfaz final: vista imagen real de caja sin piezas.	46
Figura 6.5. Imágenes de objetos de referencia.	47
Figura 6.6. Máscaras para la extracción de características locales (objeto 1: gorra).	48
Figura 6.7. N.º de puntos característicos encontrados con SIFT y SURF con y sin fuente de luz externa.	48
Figura 6.8. Emparejamientos encontrados con SIFT y SURF para imágenes con y sin <i>flash</i> .	49
Figura 6.9. Resultados de la identificación de objetos.	50
Figura 6.10. Resultados de la identificación de objetos ocultos parcialmente.	50
Figura 6.11. Identificación de objetos ocultos parcialmente con y sin fuente de luz.	51
Figura 6.12. Identificación de objeto real con SIFT y SURF.	51

# 1 INTRODUCCIÓN

---

**E**n el presente Trabajo Fin de Grado, como indica su propio título, se va a tratar de desarrollar un sistema de visión artificial para controlar el inventario de material en las distintas estanterías situadas en una planta de fabricación, con el fin de conocer la localización de cada unidad de material. En este primer capítulo introductorio se va a describir cuál es el objetivo del proyecto, indicando su alcance y el procedimiento llevado a cabo, así como una breve introducción a los resultados que se han obtenido. Adicionalmente, se justificará la elección del tema a tratar enmarcando el proyecto en el ámbito industrial al que se aplica y destacando las ventajas que ofrece el salto a la tecnología empleada: la visión artificial. Finalmente, se describirá la estructura del documento, describiendo así el contenido abarcado en cada capítulo.

## 1.1 Justificación

Actualmente, en el sector industrial se presenta la dificultad de mantener un control sobre la localización del material. Este es un problema muy común, sobre todo en el sector aeroespacial, donde en el montaje de aviones el volumen de piezas que se maneja es muy elevado. Cuando se desconoce la información de localización de los materiales y ocurre que no están en su lugar correspondiente cuando van a ser usados, la mayoría de las veces se pierde una gran cantidad de tiempo en buscarlos, encareciendo mucho el coste temporal, y por tanto económico, del proceso de fabricación del avión. Además, este problema no solo se presenta cuando se requiere el uso del material, sino también existen inconvenientes en su llegada a la planta. Las distintas piezas llegan almacenadas en cajas y se contabilizan manualmente, siendo este un método antiguo, ineficiente y muy costoso en cuanto a dinero y a tiempo.

Por tanto, la finalidad de este proyecto es dar solución al problema anterior, aplicando la visión artificial, con el objetivo de llevar un mejor control del inventario, mostrando así los resultados que se pueden obtener con esta tecnología. Posteriormente, este podría ser retomado para ser mejorado e implementado junto a otras tecnologías de una forma práctica.

La aplicabilidad de este proyecto no se centra únicamente en el sector aeroespacial, sino que es extrapolable a muchos ámbitos, como por ejemplo el control de los productos en puntos de venta al público, la identificación de objetos, o incluso el control del flujo en cuanto a recursos humanos.

Al comparar los trabajos realizados manualmente bajo la supervisión de una persona y un sistema de visión artificial, se obtienen las siguientes ventajas en favor de la segunda opción:

- Se produce un ahorro económico y temporal, al reducir los gastos asociados al tiempo invertido en la búsqueda de los materiales durante el proceso de fabricación.
- Se evitan pérdidas de materiales, que en algunos casos tienen un coste muy elevado.
- Se mejora el servicio al cliente, aumentando la capacidad productiva, siendo capaces de entregar un producto en un tiempo menor.
- Se reducen las tareas administrativas, ya que la automatización del control y la búsqueda de materiales evita la necesidad de realizar manualmente un inventario de forma mucho más frecuente.

## 1.2 Objetivo

El presente Trabajo Fin de Grado se ha llevado a cabo para la empresa aeroespacial AIRBUS, gracias a la beca XPLOER. Como se comenta en la sección anterior, su interés nace de la necesidad de reducir tiempos en el proceso de fabricación de aviones, puesto que se detecta que el hecho de no tener correctamente localizados los materiales que se requieren para el proceso productivo es un gran inconveniente, que aumenta

considerablemente el tiempo de proceso total en la fabricación de aviones.

Por ello, este proyecto trata de resolver parte del problema de localización de material dotando a las estanterías de visión artificial. Para esto se ha supuesto que las estanterías de almacenaje constan de un computador y una cámara en cada posición correspondiente a una caja de material. Se trata por tanto, de obtener los datos digitales de los materiales que contiene cada caja en cada estantería para ser enviados en un futuro a la red de *IoT (Internet of Things)* de la empresa. De esta forma se consigue tener la información relativa a la disponibilidad de las piezas, pudiendo determinar gracias a ello cuáles se tienen en cada estación de trabajo y cuáles faltan.

Como se ha comentado, el problema se resuelve parcialmente. Esto se debe a que no se consigue indicar dónde se encuentran las piezas en caso de no estar en las estanterías. Para ello se requiere un seguimiento previo, por lo que se deberían aplicar los mismos métodos en otras partes del proceso de transporte de los materiales, para así conseguir una información más detallada sobre cada pieza, determinando no solo si está disponible o no, sino también su localización exacta.

En cuanto al desarrollo del proyecto, se ha realizado con el lenguaje de programación Python y la librería OpenCV, que proporciona las funciones de visión artificial al proyecto.

Respecto a la detección de los materiales en las estanterías, se ha realizado usando cajas abiertas silueteadas con fondo rojo, que a su vez contienen información a través de códigos QR. Cuando una caja se coloca en una estantería, mediante técnicas de procesamiento de imágenes se localizan y detectan los materiales que contiene. En concreto, esto se realiza a través de dos métodos: búsqueda de características locales y conteo de área rellena por procesamiento de imágenes. Los datos obtenidos se indican en una interfaz gráfica diseñada para ejecutarse en el ordenador de cada estantería, realizada también con el mismo lenguaje de programación (Python), con el fin de que los operarios de la empresa tengan un acceso simple e intuitivo a la información de localización de la pieza que requieran.

En los resultados obtenidos mediante la aplicación del procedimiento mencionado, se muestra una posible solución válida para localizar los materiales de manera automática. Como desventaja, este método podría fallar cuando las condiciones de la captura de imagen por las cámaras son deficientes, por lo que sería posible mejorar notablemente el proceso propuesto mediante la adición de más alternativas de detección con RFID.

### 1.3 Estructura del documento

Este documento se divide en 7 capítulos, cuyo contenido se detalla a continuación:

- Capítulo 1: Introducción.

En este primer capítulo se describe el propósito del presente trabajo. De esta manera, se introduce por primera vez el tema a tratar, su aplicabilidad e interés de estudio, las herramientas y técnicas empleadas en su resolución, así como un breve adelanto de las conclusiones obtenidas.

- Capítulo 2: Localización de Materiales por Visión Artificial.

Trata de enmarcar y unir los dos temas principales del proyecto: el control de inventario y la visión artificial. De esta manera se realiza un recorrido de sus avances y su estado actual, se explican las bases teóricas más importantes y se exponen los retos que se pretenden conseguir.

- Capítulo 3: Descripción formal del problema.

En este capítulo se describe el problema desde el punto de vista del empleo de visión artificial para solucionar problemas encontrados en la producción del sector aeroespacial. Se hace analizar las aplicaciones más interesantes de esta tecnología para solucionar los problemas que se presentan, seleccionando uno de ellos para acotar el alcance del proyecto, justificando su elección y detallando sus características.

- Capítulo 4: Descripción de la metodología.

En su contenido se mencionan las metodologías existentes para solucionar problemas similares y se explican con mayor nivel de detalle las elegidas para la resolución del problema actual. Se comenta también el software elegido para su desarrollo.

- Capítulo 5: Desarrollo de la aplicación.

Este capítulo abarca la aplicación de la metodología seleccionada en el capítulo 4. De esta forma, se aborda un rediseño del entorno físico del problema y se describen los programas desarrollados para la empresa.

- Capítulo 6: Resultados experimentales.

En esta sección se muestra la apariencia del programa desarrollado y se comentan los resultados experimentales que se han obtenido para diferentes situaciones. Se comparan los distintos métodos usados para comparar el funcionamiento y efectividad de cada uno de ellos.

- Capítulo 7: Conclusiones y líneas futuras.

Por último, se resumen las conclusiones obtenidas y se presentan las posibles líneas futuras de desarrollo, de forma que pueda ser continuado por la empresa.

El documento finaliza con las referencias bibliográficas que han servido de fuente de información para la elaboración del presente Trabajo Fin de Grado.



# 2 LOCALIZACIÓN DE MATERIALES MEDIANTE VISIÓN ARTIFICIAL

Este capítulo pretende poner en contexto al lector sobre los temas que se van a tratar en el proyecto. Se empezará exponiendo brevemente la historia de la localización de materiales para entender los problemas y necesidades que surgen con relación a ella. De todas las herramientas que se disponen en la actualidad, este documento se centra en dar una solución desde el campo de la visión artificial, que cada vez ha ido tomando más importancia en la industria actual. Por ello, se comentarán brevemente los conceptos básicos de esta tecnología que serán la base del proyecto. A continuación, se mostrarán ejemplos de aplicaciones industriales que usan la visión artificial, nombrando a empresas destacables en la actualidad, como por ejemplo Amazon. Finalmente, se concluye identificando los retos que supone la implementación de este campo de la tecnología que cada vez intenta integrarse más en aplicaciones industriales.

## 2.1 Antecedentes de la localización de materiales

La localización de materiales en la industria está muy ligada a la gestión de almacenes y el manejo de inventario, ya que aporta información tan importante como es el dónde se encuentran los elementos, cuánto tiempo han estado en cada lugar, y si están disponibles para su utilización.

El origen de los inventarios se sitúa en una etapa muy antigua de la historia, puesto que tiene lugar cuando los egipcios almacenaban grandes cantidades de alimentos para afrontar los periodos de sequía. Así se aseguraba la subsistencia de la vida y un normal desarrollo de las actividades cotidianas. De ello deriva la necesidad de llevar un registro, distribución y cuidado de los recursos.

En América se conocen sus antecedentes por el año 2.500 A.C., con la creación de una herramienta llamada Quipu por el pueblo Inca, ver Figura 2.1, que hacía uso de registros, contabilidad de cosechas, reservas de producción agrícola, etc., por medio de cuerdas y un lenguaje de nudos. (Cabrera et al., 2007).



Figura 2.1. Quipus Incas. [Fuente: National Geographic en Español (2018)]

Con el paso del tiempo, es en la Segunda Guerra Mundial cuando, gracias a la aparición de grandes y complejas computadoras, surgen los primeros sistemas para la planificación de los materiales. Más tarde, de la mano del avance de la tecnología aparecen sistemas de manejo de inventarios, aumentándose así el número de empresas que hacían uso de estas herramientas.

Han tenido lugar gran cantidad de avances en lo relativo al procedimiento de cómo almacenar y contabilizar los materiales existentes. No obstante, hay casos en los que también es necesario saber dónde se encuentran en cada

momento, y situarlos desde que llegan hasta que se almacenan, puede generar muchos problemas que conducen al desconocimiento de la ubicación del inventario. Algunos de los fallos más comunes son:

- Errores y omisiones a lo largo del proceso administrativo, que producen diferencias en los valores de los inventarios.
- Desconocimiento de los movimientos que realiza el material una vez entra a las instalaciones, trazabilidad inexistente.
- Olvidos a la hora de utilizar el material y no colocarlo de nuevo en su lugar correspondiente.

Estos problemas causan enormes pérdidas de tiempo por la búsqueda del material a la hora de realizar una actividad que requiera dichos elementos y estos no se encuentren en su lugar correspondiente.

Es entonces cuando la localización de materiales juega un gran papel, ayudando a solucionar los errores anteriormente comentados. Para ello existen distintas tecnologías que se usan normalmente en la industria como son las etiquetas RFID, los códigos QR o la visión artificial, entre otros.

Lo más utilizado habitualmente son las tarjetas RFID, que son colocadas en cada pieza y leídas con receptores móviles o fijos. Sus principales ventajas son su rapidez, facilidad de uso y su precisión. Y su mayor desventaja es la necesidad de etiquetar cada material por individual y el apantallamiento magnético que puede darse por culpa de estanterías o piezas metálicas.

La tecnología que se aplicará en este documento es la visión artificial, ya es una forma de automatizar el proceso de forma barata y eficaz, siendo fácil incluir nuevos materiales para ser detectados sin necesidad de etiquetas. Además de detectar los objetos, se pueden leer códigos de barras, textos y códigos QR.

La visión artificial se va adentrando cada vez más en la industria gracias a los grandes avances en el campo de las redes neuronales que pueden ser ejecutadas incluso en pequeñas computadoras portátiles como son las Raspberry o los netbooks. Tienen la ventaja de poder solucionar problemas muy distintos con el mismo hardware y tienen la posibilidad de actualizar y mejorar los algoritmos sin necesidad de compras extras.

A continuación, se va a explicar con mayor nivel de detalle la teoría necesaria para entender los métodos de la visión artificial aplicables a la localización de materiales.

## 2.2 Visión artificial para la detección de objetos

La visión artificial incluye una gran cantidad de métodos para adquirir, procesar, analizar y comprender las imágenes tomadas del mundo real, con el objetivo de generar información tratable por un ordenador. Se pretende de esta manera, que el ordenador pueda actuar según convenga en una determinada situación después de comprender las imágenes de la manera que lo haría una persona.

La adquisición de los datos se puede conseguir por distintos medios como son la secuenciación de imágenes, datos multidimensionales con sensores LIDAR o vistas desde distintas cámaras de vídeo. Esta depende de la aplicación que se vaya a abordar que pueden ser tan variadas como la reconstrucción 3D, el aprendizaje automático o la detección de objetos de la que trata este documento.

La detección de objetos es una parte de la visión artificial que trata de detectar los objetos encontrados en una imagen y la localización de estos por su apariencia visual. Así, se pueden destacar tres métodos para realizar esta tarea que son el procesamiento de imágenes, la extracción de características 2D y las redes neuronales.

En este proyecto se hará uso exclusivamente de los dos primeros debido a que las redes neuronales necesitan de un volumen de datos de imágenes de entrenamiento muy elevado y son muy sensibles a los datos de entrenamientos para objetos de geometrías similares. Otros inconvenientes serán comentados más detenidamente en el capítulo 4, donde se van a desarrollar todas las metodologías.

Antes de adentrarse en estos métodos, es aconsejable entender los conceptos relacionados con las imágenes digitales y la visión. A continuación, se dedica un apartado a explicar la base teórica de esta rama científica.

### 2.2.1 Similitudes entre la visión artificial y la visión humana

La visión artificial permite dotar a las máquinas de la capacidad de ver el mundo que las rodea, de manera que puedan extraer información, y deducir la estructura y propiedades del mundo tridimensional mediante imágenes bidimensionales.

Estas imágenes son el punto de partida de todo sistema de percepción visual. El proceso de captar la imagen se realiza mediante algún sensor que transforme un entorno tridimensional en una representación bidimensional. En el caso de los seres humanos, la equivalencia con estos sensores son los ojos, que son capaces de captar la información visual para obtener impulsos nerviosos que el cerebro es capaz de interpretar. Para realizar esta tarea, el ojo humano se compone de partes bien diferenciadas, muy similares a los componentes que usa una cámara para obtener una imagen. Entre estas partes se destacan:

- El iris y la pupila, que se asemejan al diafragma de una cámara, encargado de controlar la cantidad de luz que entra por el sensor. En condiciones de mucha luz, el iris se contrae reduciendo la cantidad de luz que llega a la retina y en caso contrario el iris se relaja.
- El cristalino, similar a la óptica que se encarga de modificar el enfoque de los objetos en función de la distancia.
- La retina, que representa el sensor de la cámara. Es el que genera la imagen digital asignando un valor de tensión a cada píxel, proporcional a la cantidad luminosa recibida.

En ambos casos, humanos y máquinas, la luz atraviesa la pupila (diafragma), para después pasar al cristalino (óptica) y finalmente llegar a la retina (sensor).

Para formar la imagen, la retina contiene dos tipos de células receptoras que recogen diferentes aspectos del espectro de luz solar, para transformarlos en una respuesta de impulsos eléctricos que se envían al cerebro a través de nervios ópticos. Estas células se muestran en la Figura 2.2 y son las siguientes:

- Bastones: captan la intensidad luminosa, por lo que aporta aspectos como el brillo y el tono.
- Conos: discriminan la longitud de onda, por lo que son los responsables de la visión del color. Hay tres tipos de conos, y cada uno es sensible a un color: rojo, verde o azul.

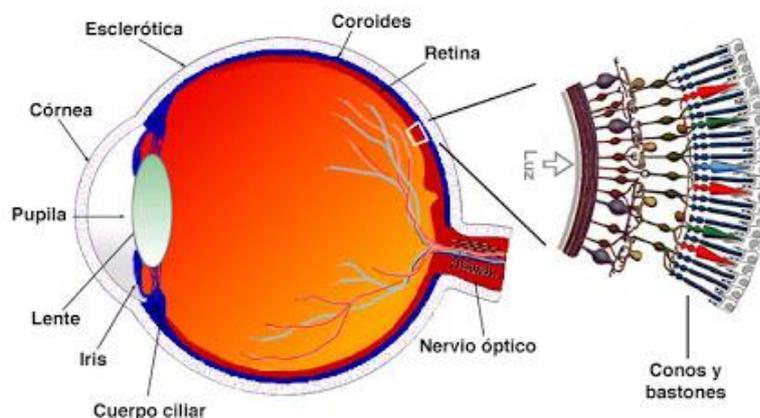


Figura 2.2. Partes del ojo humano. [Fuente: BCM Families Foundation (s.f.)]

## 2.2.2 Imágenes digitales

En las cámaras, la luz llega finalmente al sensor, que se compone de diminutos elementos sensibles a la luz que forman una matriz. Cada uno de ellos almacena la carga proporcional a la cantidad de fotones incidentes.

Para captar el color, se emplean unos filtros RGB de tal manera que cada punto discrimine el rojo, azul o verde. Esto se puede observar en la Figura 2.3, donde se ve que el patrón seguido es la matriz de Bayer, que contiene el doble de receptores verdes ya que el ojo es más sutil con este color.

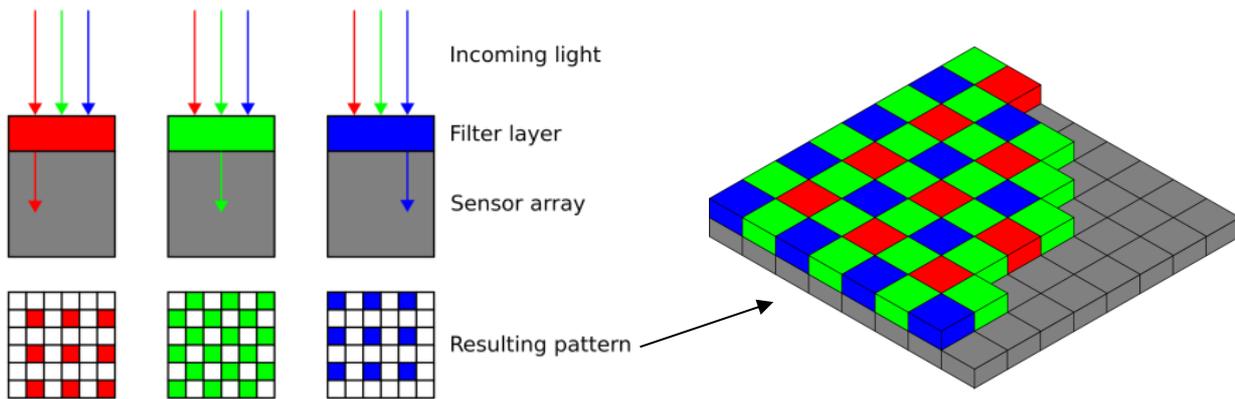


Figura 2.3. Composición de filtros para separación de fotones según su longitud de onda.

[Fuente: Adaptación del temario de “Sistemas de percepción”, Escuela Técnica Superior de Ingeniería]

Una vez explicado cómo se recoge la luz a través del sensor, se verá cómo se representa la imagen digital, y para ello hay que entender la unidad básica: el píxel.

Un píxel es la menor unidad homogénea que representa una imagen digital. Estos son pequeños cuadros pueden estar en color o en escala de grises, siendo la profundidad del color 24 bits o 8 bits, respectivamente. Esto significa que cada píxel puede tomar un color de entre  $(256 \cdot 256 \cdot 256)$  combinaciones. Esta cuantización es consecuencia de la discretización de la intensidad lumínica captada por el sensor.

Un ordenador entiende una imagen en escala de grises como una matriz de dos dimensiones, donde cada elemento es del tipo “uint8” que representa un número entero sin signo almacenado con 8 bits de información. Para una imagen en color se usan 3 matrices del mismo tamaño que la anterior, cada una representando cada uno de los colores primarios, rojo, verde y azul. De esta forma, se pueden procesar las imágenes cambiando los valores de cada elemento de las matrices que la componen. Esta manipulación de las intensidades de los píxeles de la imagen puede hacerse a través de dominio espacial o frecuencia.

Las operaciones en el dominio espacial modifican los elementos de la matriz en función de sus vecinos a través de una máscara, también llamada “kernel”. Una máscara es una matriz de dos dimensiones mucho más pequeña que la de la imagen, típicamente cuadrada y con tamaño impar. Esta tiene diferentes valores según el procesamiento que se quiera aplicar, y se usa mediante una operación de convolución trozo de la imagen original para todo su tamaño. En este tipo de técnicas se basa todo el tratamiento de imágenes como el suavizado, el aumento de nitidez, etc.

## 2.2.3 Histograma

El histograma es una gráfica usada en estadística, que aplicada a imágenes permite sistematizar algunos tratamientos. Esta es una representación de la cantidad de veces que aparecen los diferentes niveles de gris en la imagen, por ello entre sus principales aplicaciones se encuentran el cambio del brillo y contraste o la separación de regiones. En la Figura 2.4 puede verse la comparación de un histograma de una imagen poco contrastada frente a otro de la misma imagen con más contrastada.

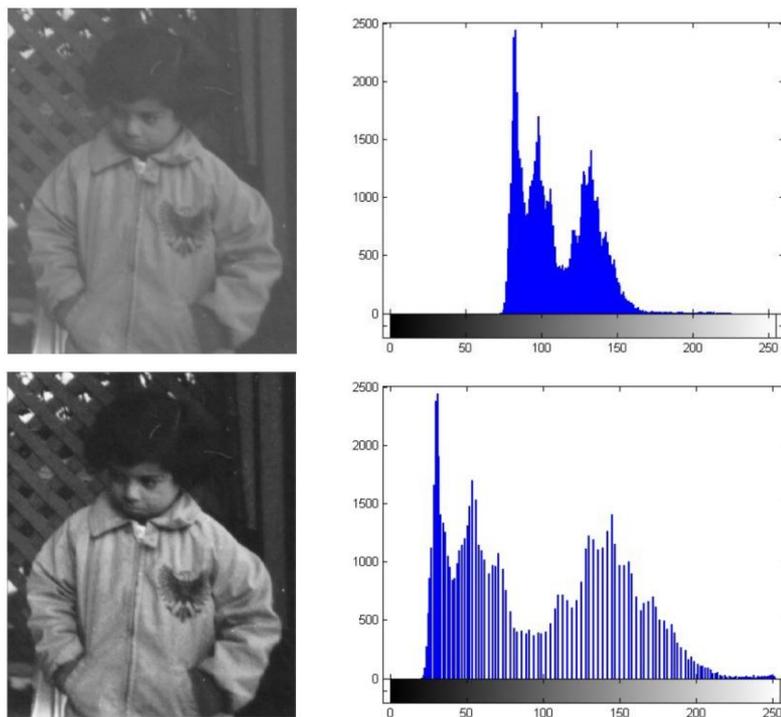


Figura 2.4. Comparación de histogramas para imágenes con diferentes contrastes.  
 [Fuente: Adaptación del temario de “Sistemas de percepción”, Escuela Técnica Superior de Ingeniería]

## 2.3 Estado del arte

Actualmente, muchas empresas hacen uso de la visión artificial para dar solución al control de stock, y entre ellas ha destacado Amazon con la creación de sus nuevos supermercados “AmazonGo”.

La peculiaridad de estos establecimientos es que las personas pueden coger productos y marcharse sin la necesidad de pasar por las líneas de caja, ya que el pago se realiza de manera automática. Para ello hace uso de la visión artificial, junto con tecnologías de aprendizaje automático (*Machine Learning*) e información de diferentes tipos de sensores, como los de profundidad o infrarrojos. De esta forma, reconocen los productos que se cogen, los que se dejan en las estanterías de nuevo e incluso informa de la falta de ellos.

A pesar de ser una aplicación prometedora tiene algunas dificultades en algunos casos como cuando se devuelven productos en distintas estanterías.

### 2.3.1 Retos de la Visión Artificial

- **Retos tecnológicos.** Hoy en día podemos encontrar muchos sistemas avanzados de visión artificial que anteriormente eran irrealizables. Se han realizado grandes mejoras en cuanto al reconocimiento en condiciones adversas, como lluvia u oscuridad, y actualmente, la visión artificial sigue en constante mejora. Uno de los retos presentes en la actualidad, es la reducción de la gran cantidad de imágenes que son necesarias para entrenar los sistemas de inteligencia artificial. Los últimos progresos en los algoritmos permiten una mejora en el rendimiento o un aprendizaje más sofisticado en el caso de redes neuronales, con una menor necesidad de imágenes para entrenar a estos sistemas.
- **Retos normativos.** La inteligencia artificial necesita de un contexto normativo adaptado a ella ya que, aunque ya han empezado a aparecer algunas normas, todavía queda mucho camino por recorrer. En el caso de la visión que se alimenta de imágenes para su funcionamiento, hay que tener en cuenta la privacidad de las personas, por lo que se requieren técnicas como el filtrado de información. Por ejemplo, en el caso de tomar fotos de personas, se debe llevar a cabo posteriormente un pixelado del rostro. También hay que tener en cuenta el almacenamiento de las imágenes tomadas ya que en la Ley

Oficial de Protección de Datos (LOPD) o en el Régimen general de protección de datos (GDPR), las exigencias varían dependiendo del tipo de datos que se guardan.

- **Retos Organizativos.** Es uno de los retos que poco a poco está consiguiéndose y es el rechazo de las empresas a utilizar esta tecnología, normalmente originado por el desconocimiento. Generalmente, se desconoce el potencial que puede tener y su capacidad de solventar problemas tradicionales que surgen en las operaciones cotidianas. Conocer el potencial de esta tecnología es el primer paso para lograr su adopción. También ocurre que, en el pasado, su implementación era compleja debido al estado de la tecnología en aquel momento, tanto por hardware como por software, por lo que no poseía la suficiente madurez como para conseguir resultados satisfactorios. La visión consigue mejorar la calidad de los procesos en los que se implementa, ayudando al operario a realizar su actividad con un mejor consumo de tiempo y esfuerzo. El número de personas dedicado a una actividad no tiene por qué verse afectado, sino que el verdadero cambio es la mejora de la calidad del proceso, no el reemplazo total de los trabajadores. A pesar de todo esto, ya se van viendo cambios en las empresas que son más reticentes a esta tecnología. Ejemplos claros son la llegada de la Industria 4.0, relacionada con la acumulación creciente de enormes cantidades de datos (*Big data*), el procesamiento de estos y la interconexión de sistemas (*IoT*), entre otros. En las industrias ya se empiezan a ver áreas de transformación digital dedicadas a la mejora de los procesos actuales, que siguen siendo manuales en muchos casos, y que ocasionan una mayor dificultad al no disponer de los datos de los procesos necesarios para evaluar la situación y realizar cambios.

# 3 DESCRIPCIÓN FORMAL DEL PROBLEMA

---

Una vez introducido el tema principal del proyecto y definidos algunos conceptos, es momento de describir el problema que se va a abordar. En este capítulo se verán las diferentes aplicaciones que se pueden solventar con la visión artificial en la empresa AIRBUS, y se detallará la que se abordará en este proyecto. Se explicará pues el origen del problema y las consecuencias que conlleva, y entre todas las distintas posibilidades de abordarlo, se seleccionará una de ellas: la localización de materiales.

En primer lugar, cabe mencionar que, para dar un salto a la transformación digital, AIRBUS se propuso a establecer un área específica para la creación y el desarrollo de soluciones usando las tecnologías en auge. Entre esta tecnología se puede encontrar, entre muchas otras, el Internet de las Cosas (*IoT*), la impresión 3D, la visión artificial, la inteligencia artificial, o la robótica manipuladora. Este proyecto nace del amplio abanico de posibilidades que ofrecen estas tecnologías, centrándose en las soluciones que puede aportar la visión artificial.

## 3.1 Elección del problema

El sector aeronáutico está en continuo avance, y cada día se intenta mejorar la tecnología que envuelve a la construcción de estos medios de transporte. No obstante, durante el proceso de montaje existen todavía algunos aspectos donde se pierde mucho tiempo de operación, como es en la búsqueda de materiales previa a cualquier actividad, o donde la intervención de una máquina facilitaría la ejecución de las tareas enormemente.

Este proyecto se centra en aportar soluciones usando la visión artificial. En la siguiente lista se recogen algunas de las posibles aplicaciones donde puede hacerse uso de esta tecnología en el sector industrial aeroespacial:

- **Identificación de materiales de montaje.**

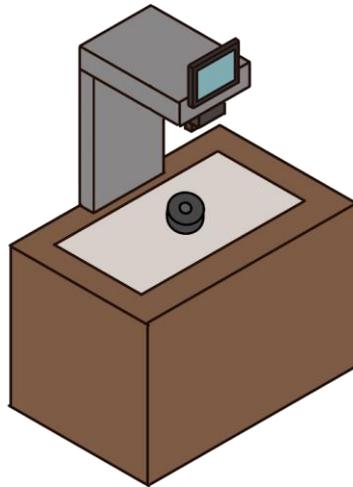


Figura 3.1. Diseño propio de sistema identificador de materiales. [Fuente: Elaboración propia]

La construcción de un avión requiere un volumen elevado de piezas, herramientas e instrumentos diferentes; y no siempre se conoce cuál se debe usar concretamente para cada operación. Cuando un trabajador debe realizar una tarea con ciertos materiales, solo ve una lista de referencias que debe de buscar manualmente en las cajas de las estanterías. A pesar de parecer una tarea sencilla, hay ocasiones en las que hay materiales en lugares que no pertenecen, o incluso ocurre que tenemos un material que no sabemos que referencia tiene, ni dónde ni cómo debe de usarse.

Para solucionar este problema, puede hacerse uso de la visión artificial con un sistema de cámaras que determina de qué material o herramienta se trata. Un posible diseño puede verse en la Figura 3.1. Esto puede resultar de mucha ayuda ya que con esta identificación se puede dar más información al operario indicando en qué lugar se almacena, en qué parte del avión se instala, en qué operación y día hay que hacerlo, etc. Este sistema puede ser integrado en un asistente virtual situado en las instalaciones, como puede verse en el siguiente apartado.

- **Identificación facial para consultas personalizadas en robots colaborativos o asistente virtual.**

Cada día es más común ver cómo más empresas se suman al uso de asistentes virtuales en páginas web, o instalados de manera física en sus instalaciones, véase la Figura 3.2. Estos ayudantes suelen recoger *cookies* en nuestra navegación que se traduce en trozos de información que les sirve para dar respuestas más personalizadas. Los esfuerzos en este campo consisten en obtener cada vez respuestas más personalizadas y útiles para la persona que utiliza el servicio. Es aquí donde aparece la visión artificial que puede aportar información extra a la búsqueda de las respuestas. En el caso empresarial, cualquier duda propuesta al asistente por un operario puede ser sesgada por su foto, detectando en qué departamento trabaja esa persona o incluso identificando materiales y herramientas de las que se necesita información a través de cámaras. Un posible diseño de esta tecnología se puede apreciar en la Figura 3.3.

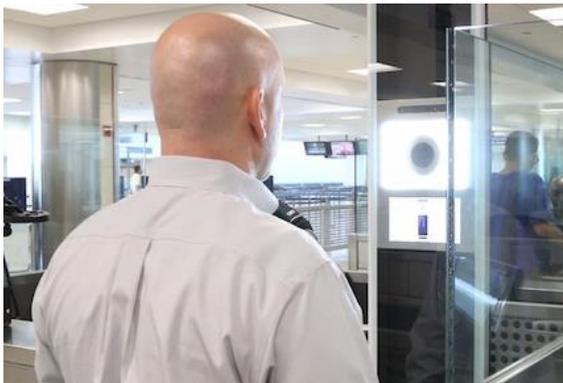


Figura 3.2. Sistema de reconocimiento facial en aeropuertos. [Fuente: (Europa Press, 2019)]

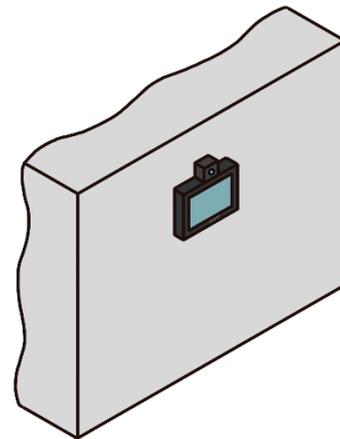


Figura 3.3. Diseño propio asistente virtual con reconocimiento facial. [Fuente: Elaboración propia]

- **Comprobación del correcto uso de las EPIs.**

La seguridad de los trabajadores es fundamental en la industria, y para ello se diseñan planes para evitar los riesgos existentes en la mayor medida posible. Existen casos en los que para reducir la posibilidad de accidente es necesario el uso de equipos de protección individual (EPI) que eliminan o reducen los daños que se pueden producir. Por el contrario, hay momentos en los que los operarios olvidan el uso de estos aumentando el riesgo de accidente. En este caso, con la visión artificial pretende hacer de vigilante automático con la ayuda de avisos sonoros, indicaciones lumínicas o incluso restringiendo el paso de los operarios a ciertas zonas si no llevan sus EPIs correspondientes. En la Figura 3.4 se muestra este tipo de sistema.



Figura 3.4. Comprobación del uso de EPIs con visión artificial.

[Fuente: Adaptación de AllGo Vision (s.f.)]

- **Control del flujo de personas para la optimización de la distribución del espacio.**

Normalmente, cada operario ejecuta las operaciones en un entorno más o menos delimitado, pero existen muchas acciones en las que se debe desplazar, ya sea en la búsqueda de materiales en el almacén, en el cambio de herramientas, o incluso cuando es hora de tomarse un descanso. A priori, puede parecer que estos desplazamientos ocasionan poca pérdida de tiempo, pero resulta que la suma de este tiempo por cada persona cada día se traduce en una pérdida de dinero considerable. En estos casos, la visión aporta información sobre el flujo de personas, permitiendo el cálculo de tiempos de desplazamiento, direcciones en el movimiento, lugares más frecuentados, etc. Con estos datos se puede hacer una optimización de la distribución de las áreas de una empresa, y así reducir los costes y el tiempo de fabricación de los productos. Estos sistemas ya se usan actualmente en las ciudades para el rediseño de las calles, véase la Figura 3.5.

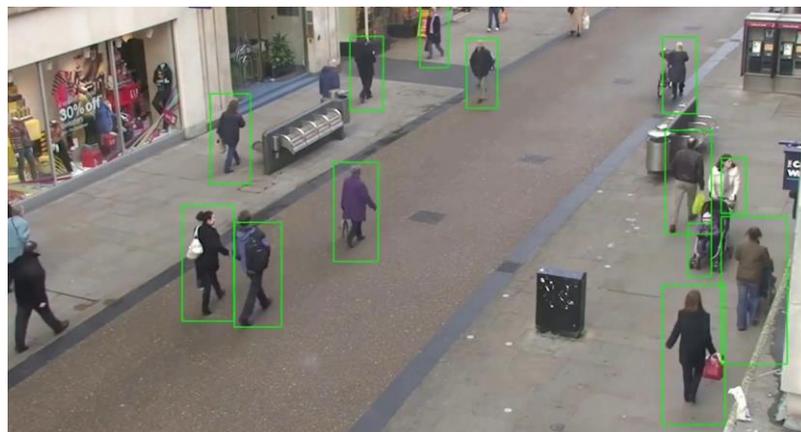


Figura 3.5. Control de flujo de personas para el rediseño de las calles. [Fuente: (Adel, E., 2019)]

- **Identificación de elementos montados en el avión.**

Un avión se construye de una enorme cantidad de componentes mecánicos y electrónicos, tantos que a veces resulta muy difícil identificarlos una vez instalados. El operario debe realizar una tarea manual de inspección y debe reconocer los elementos que se han montado para verificar las operaciones, y estas tareas pueden consumir mucho tiempo. Sin embargo, con el uso de la visión artificial se podría hacer de manera casi automática, con la detección de elementos específicos, ya sea con el entrenamiento de redes neuronales, o por uso de etiquetas con códigos QR que no afecten magnéticamente a otros elementos instalados. Además, esta solución puede venir acompañada de una tecnología de realidad

aumentada tanto para la inspección como para el aprendizaje del nuevo personal, haciendo este proceso mucho más rápido e interactivo. Una solución de este tipo podría verse en la Figura 3.6.



Figura 3.6. Realidad aumentada con visión artificial para identificación de elementos.  
[Fuente: (Neosentec, 2017)]

#### - Verificación de tareas.

Los fallos en la realización de las tareas resultan casi inevitables debidos al cansancio, la repetitividad de las acciones, u otros muchos factores. Los errores en la fabricación de un avión pueden conllevar retrasos en la entrega del producto, o incluso acabar en accidentes y catástrofes en los peores casos. Con la visión por ordenador se pueden realizar tareas de verificación, ya sea por métodos de procesamiento de imágenes o combinado con redes neuronales. Con esta tecnología se pueden verificar muchos tipos de operaciones, como es una buena soldadura o un desperfecto de las piezas, entre otros (véase Figura 3.7). Además de verificar operaciones, también se puede usar la visión para detectar si se están usando las protecciones de los aviones, que son unas cubiertas de color llamativo que se usan para evitar que entre polvo y suciedad.

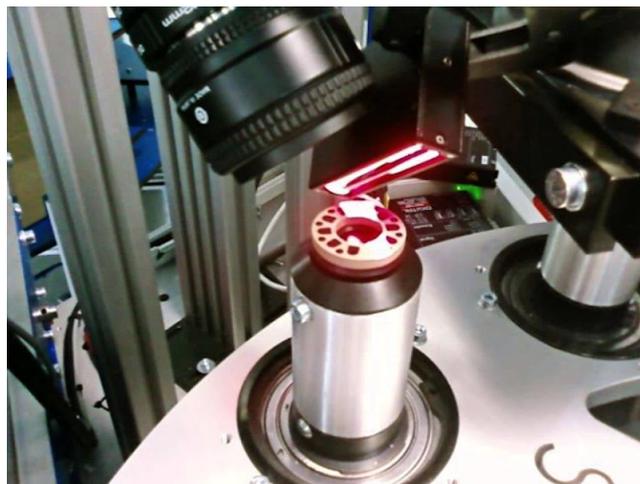


Figura 3.7. Inspección de alta precisión con visión artificial. [Fuente: (EASYCONTROL, S.L., s.f.)]

#### - Digitalización de instrumentos analógicos.

Resulta una gran inversión para una empresa comprar herramientas e instrumentos para realizar operaciones necesarias sobre el producto. Es complejo determinar cuándo es un buen momento para

actualizarlos, ya que cada poco tiempo aparecen nuevos y mejorados. Es por ello por lo que sería ideal poder actualizar herramientas analógicas en digitales para poder guardar y analizar esos datos en ordenadores. Una opción es abrir los instrumentos y convertir la señal analógica a digital, pero con este proceso se pierde la garantía de muchos fabricantes y puede estropear la medición si no se hace correctamente. Otra opción es hacer uso de la visión artificial, ya que, usando una cámara de forma externa, pueden traducirse esos números, o agujas de los visualizadores analógicos, en datos digitales que pueden ser compartidos y almacenados por un ordenador. Un proceso así se observa en la Figura 3.8. De esta manera se consigue más información a analizar, que siempre es importante para llevar un buen control de los procesos.

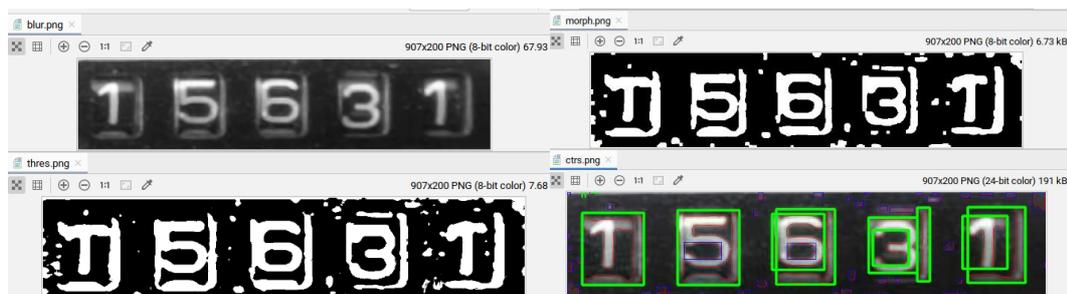


Figura 3.8. Lector de indicadores numéricos analógicos por visión.

[Fuente: Adaptación de Stack Overflow (2019)]

#### - Control y seguimiento de los materiales.

Ocurre que los nuevos materiales que entran a las instalaciones llegan directamente a un punto de recogida, donde se confirma su recepción elemento a elemento. Cabe destacar que esta es una tarea manual y tediosa en la que se pierde mucho tiempo. Después, los materiales se desplazan por la fábrica a otros puntos hasta llegar a sus estanterías correspondientes, y durante todo su recorrido no hay un seguimiento de los materiales, por lo que si no llegan a su destino no se conoce la ubicación de estos. Además, suponiendo que el material llega correctamente a su estantería, cualquier operario puede retirarlo por un momento y no volver a colocarlo en su mismo lugar. Esto acarrea muchos problemas a la hora de realizar una operación, puesto que en el sistema puede aparecer que se disponen de todos los materiales y herramientas necesarios, pero puede que la información que se muestra no se corresponda con la realidad. La visión artificial puede aportar información en varios puntos del recorrido consiguiendo conocer a tiempo la falta del material y facilitando su búsqueda, para evitar así pérdidas de tiempo.

De todas las aplicaciones expuestas, en este proyecto se aborda la última de todas, que trata sobre el control y el seguimiento de los materiales en las instalaciones. Para comprender esta decisión, se van a explicar de manera más detenida los problemas más importantes, consecuencia de la falta de seguimiento de los materiales. Posteriormente, se acotará el problema ya que se puede tratar desde distintos puntos.

## 3.2 Definición del problema

Cuando un operario va a realizar una operación, es común que en la lista que se le proporciona aparece que se encuentra todo listo para su ejecución, pero cuando va a buscar los materiales necesarios, estos no se encuentran en las estanterías. En este momento, el trabajador no tiene más información de donde se encuentran los elementos que necesita para realizar su tarea, y este tiene que buscar por los lugares más frecuentes o incluso por todas las instalaciones. En este proceso de búsqueda se emplea mucho tiempo, que se traduce en pérdidas de dinero, en parte porque el trabajador está dedicando su tiempo en acciones que no aportan valor al producto final, y también porque afecta a los tiempos de entrega de este. Estos problemas son aún más serios cuando no se consiguen localizar los materiales después de varios días.

Para entender la magnitud del asunto, se distingue la falta de material diferenciando este como:

- Material no bloqueante: aquel necesario para realizar la operación, pero su falta no perjudica a otras acciones e incluso se puede empezar dicha tarea aún con la falta de este material.
- Material bloqueante: aquel que hace falta para realizar la operación y que además su ausencia supone un bloqueo en el proceso, puesto que es parte fundamental para poder continuar con dicha operación o incluso con las siguientes. Este tipo es más perjudicial puesto que un retraso de unos días supone grandes pérdidas en la empresa.

Para abordar este problema se debe conocer primero la ruta que siguen los materiales y herramientas desde su origen, hasta que finalmente llegan a las estanterías donde se almacenan. Los pasos seguidos se pueden observar en la Figura 3.9 y son los siguientes:

1. El origen de los materiales proviene de otras fábricas que no se localizan en las mismas instalaciones de la empresa. Desde este punto, los materiales se introducen en cajas, individualmente empaquetados en una bolsa de plástico translúcida, lo que dificulta su identificación pero los protege de posibles golpes y arañazos. Posteriormente, cada caja es etiquetada con información que indica mediante referencias ciertos datos, como por ejemplo a qué avión y estación de trabajo van dirigidos. La parte más destacable es que también se indica si se encuentran todos los materiales correspondientes en la caja o si no, pero no cuál de ellos falta.
2. A continuación, los paquetes se transportan a las instalaciones de la empresa. Como se ha comentado en el paso anterior, las cajas contienen una etiqueta que indica si están todos los materiales o no. De aquí la necesidad de que un operario compruebe cada paquete recibido y haga un listado de que indique los materiales que se encuentran en las cajas y cuáles no. Como se indicó anteriormente, esta es una tarea larga y tediosa, cuya automatización supondría un gran avance.
3. Una vez comprobado lo anterior, se transportan de este espacio a otros lugares en las instalaciones sin mantener ningún control sobre la trazabilidad de dichos materiales, ni del tiempo que se quedan estacionados durante su transporte. Este punto se considera que se puede mejorar con el buen control y planificación de la ruta de los paquetes.
4. Finalmente, las cajas llegan a sus estanterías de destino correspondientes, en función de a qué estación de trabajo van dirigidas, que dependerá del producto final (avión) en el que se vayan a emplear. De las cajas se va cogiendo el material conforme sea necesario y la operación lo requiera, pero tampoco se lleva un control de cuántos elementos se sacan ni cuántos elementos quedan.

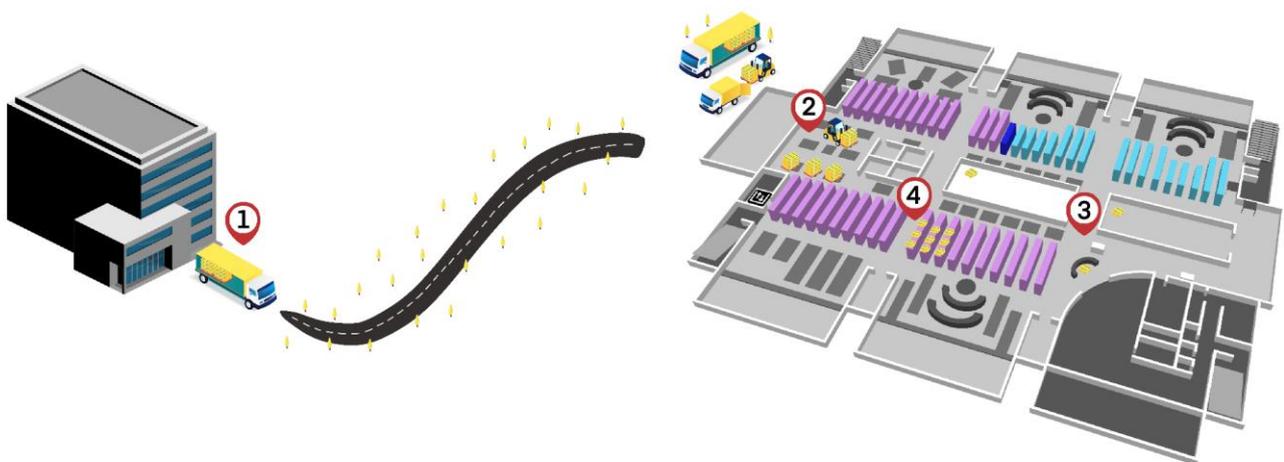


Figura 3.9 Recorrido de los paquetes de la empresa. [Fuente: Elaboración propia]

Como se puede comprobar, el material podría perderse en muchas partes del proceso de transporte.

Una vez entendido el recorrido de las cajas por las instalaciones y los problemas existentes en cada paso, se verán a continuación las razones más comunes por las que el material no se encuentra en su destino (estantería) correspondiente:

- No se ha llevado aún a la estación y se encuentra en el punto de llegada a las instalaciones.
- Se encuentra en un punto intermedio de transporte y aún no ha llegado a la estación.
- Se ha llevado a otra estantería o estación por error.
- El material llegó correctamente a la estantería de destino, pero alguien lo extrajo.
- Se ha agotado y no se había notificado.
- Errores en el inventario inicial realizado a su llegada.

A la hora de abordar este problema se encuentra con que no se disponen de datos digitales y rápidos de la localización de los materiales, y por tanto no se conoce cuáles de ellos están en el destino listos para la siguiente operación. Solo se sabe cuáles llegaron a la fábrica, pero como se ha comentado, pueden cometerse errores al hacer este inventario o puede haber pérdidas en su desplazamiento.

En el siguiente apartado se acota este problema, decidiendo qué parte del proceso de transporte se va a abordar y exponiendo las razones de esta decisión.

### 3.3 Acotación del problema

Como se ha comentado en los apartados anteriores, el problema consiste en localizar los materiales con ayuda de la visión artificial. También se ha visto que surgen varios problemas desde el origen de los paquetes hasta su llegada final a las estaciones.

Las etapas más interesantes para abordar el problema son dos: la llegada de los paquetes a las instalaciones, y la etapa final en las estanterías. Sin embargo, la magnitud de este proyecto solo permite la elección de una de estas dos. Es por ello por lo que a continuación se van a comparar las ventajas y desventajas de solucionar la localización de materiales en estos dos puntos.

En primer lugar, suponiendo que se acota el problema en la etapa de llegada de los paquetes, la solución parece obvia, y trataría de automatizar el proceso de conteo de materiales en el interior de las cajas. Para ello, existirían muchas posibilidades dependiendo de si se quiere automatizar completamente sustituyendo al anterior trabajador, o realizar un proceso que ayude al operario a realizar este conteo de forma mucho más rápida y que además los datos se digitalicen de forma automática.

Se supone que en este caso, no se podría cambiar completamente el proceso, puesto que al pasar de no usar tecnología a automatizar completamente el proceso se incurriría en un coste muy elevado. Por ello, el problema se centraría en realizar un sistema cuya entrada sea la caja, y que mediante una cámara sea capaz de detectar cuáles son los materiales que contiene y leer la referencia e información de la caja. Todo este proceso será guiado por un operario, cuya labor consistiría en introducir la caja, comprobar que se ha hecho una buena detección y aceptar la operación. Con esto se conseguiría la digitalización de los datos que pueden ser usados en muchos programas corporativos. Sin embargo, no se resuelve todo el problema con esta solución, puesto que las cajas pueden extraviarse durante el camino a las estaciones, o incluso los materiales pueden ser extraídos de las estanterías sin volverse a colocar.

En segundo lugar, la otra opción que se propone es solventar el problema en las estanterías, resolviendo así la incógnita de si los materiales realmente se encuentran disponibles en el lugar de trabajo. De esta manera se pueden crear estanterías que usen la tecnología para leer el interior de las cajas cada cierto tiempo para obtener los datos digitales de los elementos que se encuentran almacenados en las estaciones. Con este sistema también se podría detectar el momento en el que los materiales han sido extraídos, durante cuánto tiempo, e incluso ver las imágenes del interior de las cajas en tiempo real desde cualquier ordenador si esa información se comparte con la base de datos.

Se debe tener en cuenta que en las instalaciones hay distintos tipos de estanterías adaptadas para cada uso. Por lo tanto, la solución a este problema debe incluir el diseño de una estantería estándar que unifique todas las anteriores y que sirva para usar el sistema de visión artificial con los mínimos fallos posibles.

Puede intuirse que, para los dos casos, no se puede aplicar la visión artificial si las cajas están cerradas o si contienen muchos elementos aglutinados quitando visibilidad unos a otros. Por ello, para estas soluciones también se necesita del diseño de un nuevo tipo de caja adaptado a este tipo de tecnología.

Finalmente, se concluye que el problema que tiene mayor prioridad en ser resuelto es el de la etapa final en las estanterías, correspondiente a la segunda opción propuesta, puesto que la información más importante es saber con certeza si el material se encuentra todo en su lugar. Para ello se darán 3 soluciones en los siguientes capítulos:

- El diseño de una estantería estándar.
- El diseño de las nuevas cajas abiertas.
- El diseño del software usado para reconocer los materiales.

# 4 DESCRIPCIÓN DE LA METODOLOGÍA

---

**E**n este capítulo se mencionarán las diferentes tecnologías disponibles con las que se puede abordar el problema. Se hablará de la tecnología RFID y visión artificial exponiendo las ventajas y desventajas de cada una, para así justificar la elección de esta última. Luego se centrará en la visión para definir los métodos de búsqueda de materiales que se pueden aplicar, y se explicarán con detalle para entender la resolución del problema en el siguiente capítulo. Finalmente, se hablará del entorno de programación utilizado, justificando la elección tanto de lenguaje, como de programas y librerías.

## 4.1 Elección de la tecnología

En este apartado se van a comparar las diferentes tecnologías existentes usadas en la industria para resolver los problemas de control de inventario y localización de materiales. Adicionalmente, se van a definir cada una de ellas indicando las soluciones que se dan actualmente y analizando sus ventajas y desventajas.

### 4.1.1 RFID

La tecnología RFID se puede definir como un sistema capaz de identificar y localizar objetos mediante la lectura de información electrónica sin la intervención humana, evitando así posibles errores por el ingreso de datos manual. Esta usa señales de radio, por lo que no es necesario el contacto físico directo y se pueden identificar una gran cantidad de elementos en muy poco tiempo de forma fiable.

Para entender mejor su funcionamiento, se van a definir los componentes necesarios que se indican en (Khattab et al., 2017). La arquitectura de la tecnología RFID se puede resumir en los siguientes componentes:

- **Etiqueta:** se compone de una antena, un transductor de radio y un chip. La antena sirve para transmitir la información contenida en el chip que identifica a dicha etiqueta. Este chip posee una memoria interna que depende del modelo y que puede variar de una decena, a millares de bytes. Existen distintos tipos de memoria dependiendo de la modificación de su información y de la detección simultánea, que son las de solo lectura, lectura y escritura, o anticolidión. La forma más habitual de estas tarjetas puede verse en la Figura 4.1.



Figura 4.1. Etiqueta RFID. [Fuente: (Witte printed technology, s.f.)]

- **Lector:** dispositivo que utiliza señales de radiofrecuencia para detectar y obtener la información contenida en las etiquetas, ver Figura 4.2. Este envía periódicamente señales para detectar si hay etiquetas en sus inmediaciones. Cuando detecta una de ellas, extrae su información y se pasa al

subsistema de procesamiento de datos.



Figura 4.2. Lector RFID. [Fuente: Adaptación de FQ Ingeniería Electrónica (s.f.)]

- **Subsistema de procesamiento de datos:** procesa los datos y almacena la información obtenida. Es el que gestiona el intercambio de información con el sistema de gestión de la empresa.

Existen muchas aplicaciones donde se puede aplicar esta tecnología, como métodos de pago, control de préstamos en las bibliotecas o en el caso de este proyecto, la localización y control del inventario.

Actualmente la tecnología RFID es muy usada para esta aplicación, ya que es muy fiable y rápida. Otras ventajas destacables son:

- Puede leer varias etiquetas en muy poco tiempo.
- Puede automatizar el proceso con antenas receptoras que hacen de lector de etiquetas RFID.
- Las etiquetas pueden soportar un entorno hostil.
- Identificación unívoca de los elementos.
- Rango de lectura extenso.
- No requiere línea visual con las etiquetas.

Existen muchas más ventajas que justifican la gran acogida que ha tenido esta tecnología. No obstante, hay algunos casos en los que no funciona como debería. Las desventajas más importantes son:

- El apantallamiento magnético que se produce debido a los materiales metálicos que queremos localizar y al intentar leer las etiquetas a través de una estantería metálica.
- La necesidad de etiquetar todos los materiales puede resultar costoso.

Se puede concluir que es una muy buena opción ya que se obtienen resultados muy fiables y ya está muy implantado en la industria. Sin embargo, debido a la gran importancia que tienen sus desventajas, no se puede tomar únicamente este sistema para la localización de los materiales.

#### 4.1.2 Visión artificial

Como se definió en el capítulo 2, la visión artificial o visión por computador, se basa en la adquisición y procesamiento de imágenes del mundo real para producir información que pueda ser tratada por un ordenador.

Los componentes de los sistemas de visión pueden variar dependiendo de la aplicación. En este caso los componentes que se usan son los siguientes:

- **Cámara y lente:** es el dispositivo encargado de tomar la imagen para ser analizada posteriormente por la computadora. Su elección es importante ya que afecta en muchos parámetros de la fotografía, como son las dimensiones, la cantidad de luz que recoge, el tamaño de la zona que se captura, etc. Para cada

aplicación se necesitarán unas características determinadas por el entorno de trabajo donde va a implantarse.

- **Fuente de luz:** es un aspecto muy importante, puesto que las condiciones de iluminación del ambiente afectan en gran medida al resultado que se obtiene. La fuente de luz debe ser uniforme, evitando reflexiones o sombras, y a ser posible, debe resaltar los rasgos de interés para una determinada aplicación. Para el problema que se aborda en este documento, el interior de las estanterías son zonas de baja luminosidad por lo que una fuente de luz a la hora de la toma de fotos o video es imperativa.
- **Algoritmos de visión:** es el código diseñado por el programador y que permite el procesado y la extracción de información de la imagen. Un algoritmo puede aplicar varios métodos de visión, los cuales se van a ver a continuación.
- **Computadora:** se encarga de ejecutar los algoritmos de visión para el análisis de las imágenes. Dependiendo de las características de este, se podrán ejecutar los algoritmos con menor o mayor rapidez.

En cuanto a las ventajas que presentan los sistemas de visión artificial, son las siguientes:

- Pueden detectar varios elementos en poco tiempo.
- Se automatiza el proceso con las cámaras integradas en las estanterías.
- Versatilidad al poder cambiar o mejorar los algoritmos de análisis de imágenes.
- Inexistencia de interferencias.
- Sistema relativamente barato.

Sin embargo, también aparecen varios problemas a la hora de aplicar esta tecnología:

- Necesaria una línea de visión directa.
- Posibilidad de fallo a la hora de reconocer algunos elementos.

Finalmente, como se ha comentado numerosas veces en este documento, el sistema elegido es el de visión artificial ya que sus desventajas pueden ser tolerables.

Diseñar una solución dentro de la empresa y no recurrir a opciones externas de otras compañías permite ahorrar mucho dinero y a la vez tener el control del software y hardware para poder actualizarlo cuando sea necesario.

En el siguiente apartado se analizarán los diferentes métodos de visión existentes para la detección de objetos y se explicará la teoría para entender la solución aplicada.

## 4.2 Métodos de visión artificial para la localización de materiales

Los métodos que se aplican en visión para detectar un objeto son muy variados y depende en gran medida de la aplicación y del entorno físico donde se aplican. Los métodos más simples son los de procesamiento de imágenes, siendo muy útiles por ejemplo cuando hay diferencias de color entre el objeto y el fondo o cuando existen círculos o líneas. Otros métodos más complejos incluyen la extracción de características 2D o el uso de redes neuronales, donde hacen falta imágenes de referencias para clasificar los objetos de las imágenes.

### 4.2.1 Procesamiento de imágenes

El procesamiento de imágenes se puede definir como un conjunto de técnicas que se aplican a imágenes digitales con el fin de mejorar la calidad y facilitar la búsqueda de la información. Además, se puede conseguir una clasificación y segmentación de objetos usando únicamente dichas técnicas que se basan en la discretización por colores, métodos morfológicos, uso de laplacianos para detectar contornos, etc.

Este procesamiento suele venir acompañado de otras técnicas como la de extracción de características por

métodos 2D. Por ejemplo, suponiendo que se tienen que comparar los objetos situados en una superficie plana con distinto color, primero se puede hacer un procesamiento de imagen para aislar los elementos por su color, y posteriormente hacer una búsqueda de características locales con el segundo método comentado para analizar la pieza en cuestión.

En este apartado se pretende explicar y dar a conocer cuáles son las técnicas de procesamiento de imágenes más usadas en este proyecto, que son las siguientes:

- **Binarización de la imagen.**

Como se comentó en el segundo capítulo, las imágenes digitales se componen de un rango de valores de intensidad conocido como escala de grises. Los valores posibles de un píxel son 256, pero dependiendo de la aplicación pueden necesitarse menos valores para diferenciar algunos rasgos u objetos de la imagen.

La reducción de estos niveles de intensidad puede resultar en una imagen binaria, compuesta únicamente de 2 valores que comúnmente son 0 y 255, que se traducen en negro y blanco, respectivamente.

Esta reducción en la representación de la imagen consigue un mejor aprovechamiento de la memoria y del coste computacional, a la vez que sirve de base para obtener propiedades geométricas y topológicas de los objetos que se encuentran en la imagen.

La binarización se puede definir como el proceso por el cual se convierte una imagen en escala de grises, en una imagen binaria. Esta es una forma sencilla de discretizar un objeto frente un fondo o diferenciarlos del resto, y por ello, es útil en la aplicación de búsqueda de materiales.

Se pueden usar diferentes técnicas para binarizar una imagen, que van desde usar un umbral simple, hasta una binarización por histograma. A continuación, se comentan brevemente las distintas posibilidades:

- **Umbral simple:** consiste en aplicar para cada píxel un umbral numérico impuesto. Si el valor del píxel es menor que el umbral se asigna un 0, de lo contrario se asigna el valor máximo. Por tanto, es necesario que la imagen de entrada sea en escala de grises. Binarizar siguiendo esta técnica no ofrece buenos resultados en muchos casos puesto que el rango de niveles de grises no siempre se extiende de la misma forma por toda la imagen, siendo necesario usar diferentes umbrales para diferentes zonas. La función usada para ello es "*cv.THRESH\_BINARY*".
- **Umbral adaptativo.** El método anterior usaba un valor global de umbral, pero esto puede no ser bueno en todos los casos si la imagen tiene diferentes condiciones de iluminación en diferentes áreas. Este método aplica un umbral diferente a cada píxel en función de una pequeña región a su alrededor, lo que proporciona mejores soluciones en imágenes con iluminación variable. La función usada para ello es "*cv.adaptiveThreshold*". A continuación, la Figura 4.3 compara el uso de estos dos métodos aplicados a una imagen con zonas con distinta iluminación.
- **Binarización Otsu:** La clave de una buena binarización es la correcta elección del umbral que separa el fondo del objeto. El umbral ideal podría ser elegido automáticamente si se conociese la distribución de intensidades de los píxeles. Esta distribución es conocida como histograma y contiene la información del número de píxeles asociados a cada nivel de gris. Si la distribución de los píxeles claros y oscuros está suficientemente separada el histograma puede ser bimodal, ver Figura 4.4, y en tal caso el valor ideal de umbral se encuentra en el valle de estos dos nodos. La función usada es la misma que en primer método con el argumento "*cv.THRESH\_OTSU*".
- **Binarización por color:** Hay casos en los que la única manera de diferenciar un objeto es por su color. Una imagen a color se compone de tres imágenes en escala de grises que modelan el

color rojo, azul y verde. Para cada una de estas imágenes se pueden aplicar las técnicas anteriores y en conjunto diferenciar los colores deseados. Esta tarea puede resultar muy sencilla dependiendo del espacio de color elegido. El modelo de color más común para esta tarea es el HSV ya que el canal de tono H modela el tipo de color y se le puede aplicar directamente el umbral. Esta última técnica es la más interesante puesto que con la elección correcta del fondo, puede ser muy fácil diferenciar los diferentes materiales en la imagen.

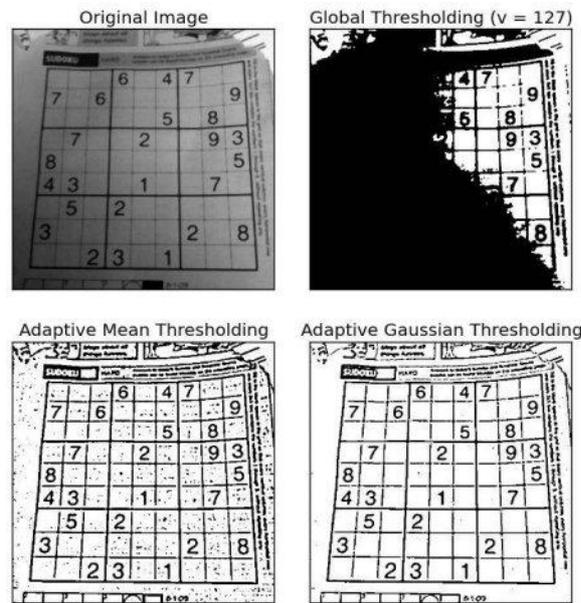


Figura 4.3. Resultados de distintos tipos de binarización. [Fuente: (OpenCV, s.f.)]

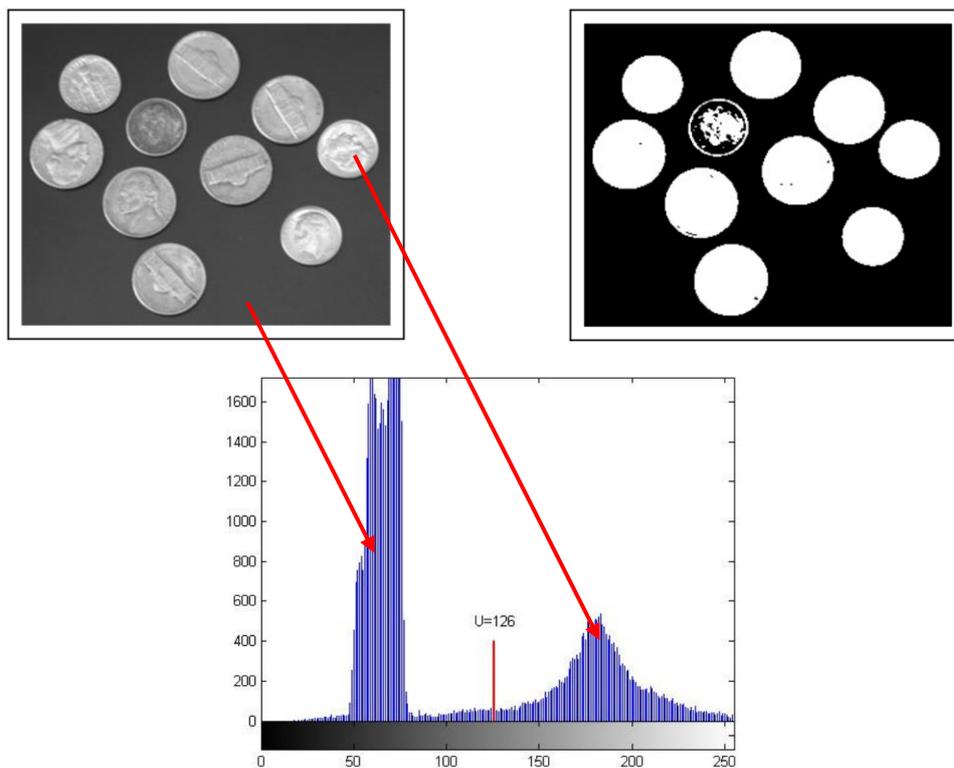


Figura 4.4. Resultado de la binarización Otsu. [Fuente: Adaptación del temario de “Sistemas de percepción”, ETSI]

### - Búsqueda de contornos.

Un contorno se puede entender como una curva que une todos los puntos contiguos que tienen el mismo color o intensidad. De esta forma, con el proceso anterior de binarización, se pueden diferenciar los objetos del fondo, pero el resultado queda en una imagen binaria con zonas negras y blancas. Para poder separar y manejar los diferentes elementos se usan métodos de búsqueda de contornos que además permiten calcular propiedades como el área, perímetro, el rectángulo que lo contiene, etc.

Las librerías usadas en este proyecto incorporan una función para realizar este método de manera automática y se trata de: *findContours*. El resultado obtenido es una lista de contornos, donde cada uno está compuesto de las coordenadas de los puntos del límite del objeto.

Una vez diferenciados los elementos se pueden realizar varias operaciones como calcular algunas propiedades o dibujar el contorno del objeto, ver Figura 4.5. Además del contorno también se pueden dibujar otras formas que contienen al objeto, como rectángulos, círculos o elipses.

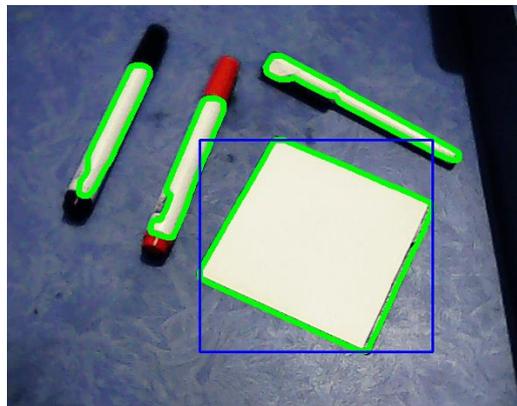


Figura 4.5. Resultado de la operación de búsqueda de contornos. [Fuente: (Rehman, 2020)]

Otras búsquedas de contorno no necesitan una previa binarización de la imagen y por el contrario basta con una entrada de esta en escala de grises. Estos operadores son: Canny y Sobel, comparados en la Figura 4.6.

- **Canny:** es un algoritmo usado para detectar los bordes de las imágenes. Es uno de los mejores métodos que existen para la detección de contornos y para ello hace uso de máscaras de convolución y de la primera derivada. Los puntos del contorno se encuentran en aquellas zonas donde existe un cambio brusco en el nivel de intensidad de los píxeles.
- **Sobel:** consigue la detección de los bordes de la imagen a través de la aproximación al gradiente de la función de intensidad. Para ello, obtiene en cada punto el vector gradiente correspondiente y la perpendicular a dicho vector. El resultado que se obtiene muestra como de brusco o suave cambia una imagen para cada punto analizado y de esta forma se muestra el borde y su orientación.



Figura 4.6. Comparativa de operador Canny y Sobel. [Fuente: Adaptación de Taibani (2019)]

#### - **Búsqueda de círculos.**

Para definir un círculo se precisan de tres parámetros que son la posición central (x, y) y el radio. Para conseguir una mayor eficiencia, el método implementado es el de gradiente de Hough, que es más complicado que los anteriores y consta de dos etapas. La primera es una detección de bordes y búsqueda de los posibles centros de círculo y la segunda es la búsqueda del mejor radio para cada candidato.

Como todos los métodos anteriores, existe una función que implementa todos los pasos de este algoritmo y que va a ser de gran importancia para la solución descrita en el siguiente capítulo.

### 4.2.2 Extracción de características locales

Los algoritmos que usan la visión artificial para reconocer patrones en las imágenes son muy variados y emplean diversas técnicas según diferentes objetivos. En este apartado se van a explicar los conceptos necesarios para entender estos métodos y se verán los algoritmos más usados para esta tarea.

Los conceptos previos pueden ser fáciles de entender si se piensa en la manera en que las personas resuelven un puzle y se proyecta esta idea a un ordenador. Cabe preguntarse pues, ¿cómo se resuelve un puzle?, ¿cómo se unen pequeños fragmentos de imágenes para formar una mayor? La respuesta es que las personas buscan una serie de patrones y rasgos específicos que son únicos y pueden ser fácilmente identificados y comparados. De esta manera se ven esos rasgos en una imagen, y buscándolos pueden identificarse fácilmente. Pero ¿cuáles son dichos rasgos característicos que pueden detectar las personas y los ordenadores? Es difícil explicar cómo una persona encuentra estas características, pero si se mira la Figura 4.7 se puede encontrar algo interesante.



Figura 4.7. Posibles rasgos característicos de las imágenes. [Fuente: (OpenCV, s.f.)]

En la parte superior de la imagen se han destacado 6 fragmentos, y se pretende responder cuantos resultados correctos se pueden encontrar para dichos fragmentos.

El A y el B son superficies planas que se pueden encontrar en gran parte de la imagen, por lo que es difícil encontrar una localización exacta.

El B y el C son bordes del edificio, por lo que se puede encontrar la localización aproximada, pero sigue siendo difícil encontrar la exacta. Esto es debido a que este patrón puede encontrarse por todo el borde del edificio. Comparándolos con los fragmentos anteriores, un borde es mejor característica que un área plana, pero aún no contiene la suficiente información para determinar la localización exacta.

Por último, el E y el F son esquinas del edificio y estas sí pueden ser localizadas fácilmente. Ya moviendo estos fragmentos por la imagen solo pueden encontrarse en un sitio específico, por lo que se puede concluir que son

buenas características.

De la misma manera se puede ver en la Figura 4.8 aún más simple.

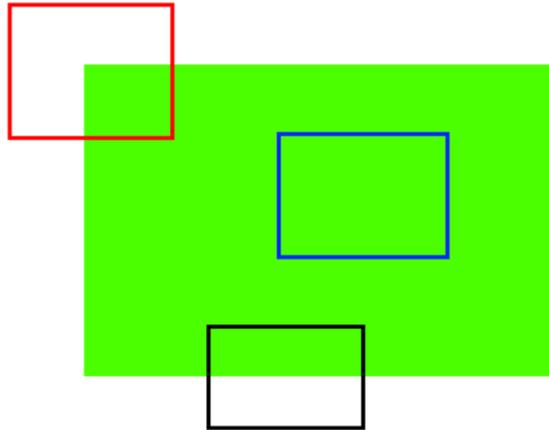


Figura 4.8. Rasgos simples de las imágenes. [Fuente: (OpenCV, s.f.)]

Una vez se sabe cuáles son las características más interesantes de una imagen cabe preguntarse, ¿cómo pueden ser encontradas? Este procedimiento de búsqueda es un concepto muy recurrido y se conoce como “*Feature Detection*”. Por ejemplo, una forma de encontrarlas es buscar las regiones que tengan las máximas variaciones a la hora realizar pequeños movimientos en todas sus direcciones. En el caso de las esquinas existen varios algoritmos como “*Harris Corner Detection*” y “*Shi-Tomasi Corner Detector*”.

Tras ser encontradas estas características, estas deben poder encontrarse en otras imágenes. Esto se hace describiendo la región que la rodea de una manera específica según los rasgos de la imagen que se estén buscando. A este método se le denomina “*Feature Description*” que ofrece como resultados los descriptores.

Los algoritmos para detectar y describir características que se usan en este proyecto son los siguientes:

- **SIFT (“Scale-invariant feature transform”)**

Algunos detectores de esquinas como “*Harris*” o “*Shi-Tomasi*” son invariantes a la rotación ya que, aunque se cambie la orientación de la imagen, la esquina sigue siendo la misma. Esto no ocurre con el escalado ya que, si la imagen es analizada con un mismo tamaño de ventana, aumentando la imagen en la esquina puede que se detecten únicamente bordes. Este efecto se observa claramente en la Figura 4.9.

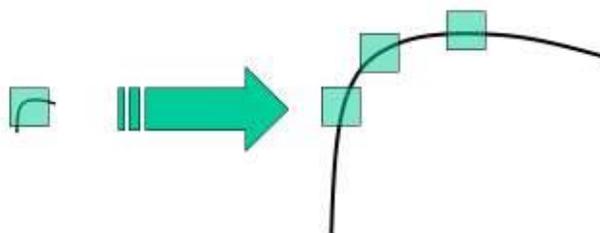


Figura 4.9. Varianza del escalado en descriptores Harris y Shi-Tomasi. [Fuente: (OpenCV, s.f.)]

Fue en 2004 cuando D. Lowe publica el algoritmo SIFT que pretende definir características y descriptores que son invariantes a ciertas transformaciones como son el cambio de escala, orientación, iluminación, etc.

SIFT es un método complejo con un alto coste computacional, pero con su correcta implementación puede ser ejecutado en muchas aplicaciones de tiempo real cuando la base de datos de búsqueda no es tan extensa. Este algoritmo está dividido en cuatro pasos que son los que se describen a continuación:

### 1. Detección de extremos en el espacio-escala.

Pretende encontrar puntos invariantes a la rotación, translación y escalado, es decir, aquellos de interés en la imagen, llamados puntos característicos. En etapas posteriores la cantidad de estos disminuirá debido a que no cumplirán algunos requisitos y se descartan. Estos puntos serán los puntos extremos que se obtienen de realizar diferencias de Gaussianas aplicadas en el espacio-escala de la imagen.

### 2. Localización exacta de los puntos característicos.

El paso anterior produce múltiples candidatos posibles entre los que se encuentran puntos con bajo contraste, que no son buenos frente a cambios de ruido e iluminación. Para descartarlos se hace uso de la serie de Taylor. Otros puntos no interesantes son los que corresponden a bordes y no a esquinas, por lo que también se aplica un algoritmo análogo al de Harris.

### 3. Asignación de la orientación.

Para conseguir características invariantes a la rotación, se le asigna una orientación a cada punto de la imagen basándose en las características locales. Esto permite poder describir puntos característicos relativos a estas orientaciones.

### 4. Descriptor del punto característico.

Para generar el descriptor del punto característico se toma una región de  $16 \times 16$  que rodea a este, y se divide a su vez en otras 4 regiones de  $4 \times 4$ . Para cada subregión se genera un histograma de orientación de 8 direcciones distintas, lo que proporciona una descripción de  $4 \times 4 \times 8 = 128$  valores.

### 5. Cálculo de correspondencias.

Finalmente, cuando dos imágenes tienen descriptores muy parecidos, es muy posible que estén describiendo la misma zona y por tanto que sean correspondiente. En la Figura 4.10 se ven los resultados de esta comparación, aplicando posteriormente un proceso de cálculo de la orientación.

Esto no siempre es así y si no se pone ninguna restricción es más probable que se obtengan muchos resultados erróneos. Además, esto produce una elevación en el coste computacional que puede provocar retrasos, muy importantes en las aplicaciones de tiempo real.

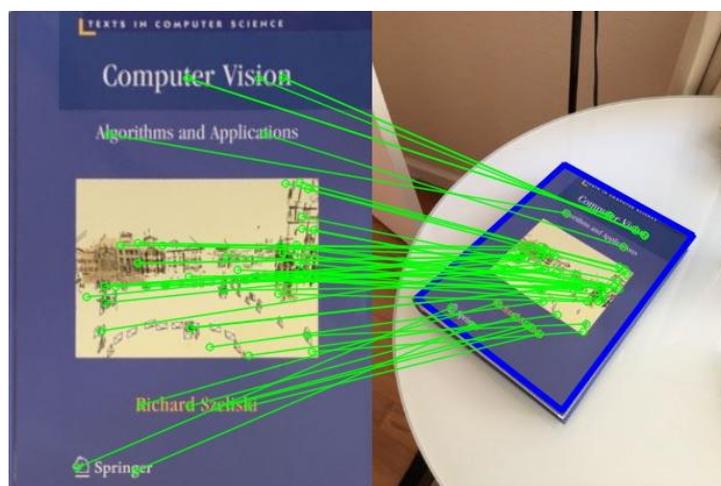


Figura 4.10. Correspondencia de puntos característicos con el método SIFT. [Fuente: (Luke, 2018)]

- **SURF (“Speeded-Up Robust Features”)**

El método SIFT es lento para algunas aplicaciones por lo que se desarrolló en 2006 un algoritmo acelerado llamado SURF, basado en SIFT. Ambos métodos son muy parecidos, pero presentan diferencias importantes. La mayor ventaja destacable de este nuevo método es la velocidad de cálculo y la robustez ante transformaciones en la imagen.

SURF se basa en la reducción de la complejidad y dimensión de los descriptores, de tal manera que continúen siendo lo suficientemente característicos como para conseguir buenos resultados.

Los pasos que sigue este método son muy parecidos al anterior, al que se le producen algunos cambios para mejorar la rapidez de cálculo.

Para la aproximación al laplaciano de Gauss antes se usaba la diferencia de Gauss. Ahora SURF usa la convolución con filtro de caja. La ventaja es que se puede calcular fácilmente con ayuda de imágenes integrales y se puede hacer en paralelo para diferentes escalas, lo que se traduce en un menor coste computacional. Para encontrar los puntos de interés se usa el determinante de la matriz hessiana, por su velocidad de cálculo y por su precisión.

Para la asignación de la orientación se usan respuestas de Wavelet, donde se otorga al descriptor invarianza ante la rotación.

Posteriormente, para la descripción de la característica, se toma una vecindad de tamaño 20x20 píxeles alrededor del punto y se divide en subregiones de 4x4, donde se calculan las respuestas de Wavelet horizontales y verticales. Con esto se forma un vector que hace de descriptor.

En resumen, SURF mejora la velocidad en cada paso y es bueno para imágenes con desenfoque y rotación, pero no es bueno para cambios de punto de vista y cambios de iluminación. Es ideal para aplicaciones donde se necesita mayor velocidad de respuesta, manteniendo buenos resultados a la salida.

### 4.2.3 Redes neuronales

Las redes neuronales están asociadas al aprendizaje automático, mayormente conocido por término anglosajón “*Machine Learning*”. Esto se basa en el entrenamiento de sistemas capaces de cambiar su comportamiento de forma autónoma en base a una experiencia, siendo capaces de aprender con unos datos dados.

Existen muchos tipos diferentes de redes neuronales, pero las más comunes para el análisis de imágenes son las llamadas redes convolucionales. Este tipo es conveniente ya que analiza no solo un punto concreto, sino también los más cercanos a este.

Como los demás tipos de redes, el objetivo de esta estructura es reducir la información de la imagen a un único valor numérico que represente la probabilidad de pertenecer a una determinada clase. Una vez diseñada, debe ser entrenada usando un aprendizaje supervisado o no supervisado. Para no complicar la explicación, si se supone que se elige el primer tipo, se necesita de un conjunto de imágenes con sus estructuras segmentadas ya asociada. Esto quiere decir que cada imagen de entrenamiento debe ser analizada manualmente, etiquetando las clases que se encuentran en ella.

Este set de entrenamientos se pasa por la red que se encargará de modificar los pesos internos de cada capa a través de un algoritmo iterativo, de forma que obtenga unas etiquetas de los datos de entrada lo más parecidas a las que hemos indicado nosotros.

Las redes neuronales ofrecen una buena solución para diferenciar algunos objetos por su clase. Sin embargo, se necesita una enorme cantidad de imágenes de entrenamiento en diferentes condiciones, llegando incluso a los cientos de miles de imágenes. Además, hay que tener en cuenta muchos factores como la calidad del conjunto, el número de iteraciones a entrenar, entre otros, haciendo muy compleja y larga la puesta en marcha de esta solución.

Para el problema de la localización de objetos en las estanterías, se pretende diferenciar entre los miles de tipos

de herramientas y piezas existentes en las instalaciones. Una red neuronal de tal magnitud no sería capaz de ajustar sus pesos internos de manera adecuada y supondría un consumo de tiempo muy grande a la hora de fotografiar y etiquetar cada objeto a clasificar.

Finalmente, tras haber analizado cada método, los más interesantes para la aplicación de visión artificial son el procesamiento de imágenes y la extracción de características locales.

### 4.3 Entorno de programación

El proyecto se ha realizado sobre el lenguaje de programación Python con las bibliotecas de visión artificial OpenCV en el entorno de desarrollo Visual Studio en su versión 2019.

#### 4.3.1 Python

El lenguaje elegido ha sido Python frente a otras alternativas posibles, como C++, Java, C# o MATLAB. La decisión se ha tomado en base a que se buscaba un lenguaje con gran capacidad computacional y simultáneamente proporcionase una sintaxis simple y clara, que permita realizar programas compactos y legibles. Estos son requisitos fundamentales, ya que se trata de un proyecto que puede ser retomado por otros desarrolladores de la empresa.

Python se define por las características que ofrece. Entre las principales se pueden destacar las siguientes:

- Multiplataforma, es compatible con numerosos sistemas operativos.
- Numerosas librerías nativas y externas ayudan a añadir gran variedad de funcionalidades de alto nivel.
- Lenguaje de programación interpretado, no hay que compilar el código antes de ejecutarlo.
- Fuertemente tipado, una variable con un valor de un tipo concreto no se puede usar como si se tratase de otro tipo a menos que se realice una conversión.
- Multiparadigma: predomina el estilo de programación orientada a objetos, pero también permite otros como la programación imperativa y programación funcional.

Además, presenta una serie de ventajas frente a otros lenguajes, que refuerzan la decisión tomada:

- Portabilidad
- Alto nivel de abstracción
- Tipado dinámico, por lo que no es necesario declarar el tipo de dato que va a contener una determinada variable, sino que su tipo se determinará en tiempo de ejecución según el tipo del valor al que se asigne.
- Una gran comunidad, donde puede encontrarse muchos tutoriales y ejemplos que facilitan el aprendizaje y la creación de programas.

Como se ha comentado anteriormente, un gran potencial de este lenguaje es el uso de librerías externas que añaden grandes funcionalidades. Las que se han usado en este proyecto son:

- *NumPy*: para cálculo numérico, da soporte a vectores.
- *PyQt5*: para la interfaz gráfica.
- *OpenCV*: añade funciones de visión artificial.

El apoyo que los desarrolladores ofrecen a este lenguaje se extiende también a sus librerías. Como la más esencial es OpenCV se ha realizado un estudio comparando las aportaciones existentes sobre esta en la plataforma GitHub para diferentes lenguajes de programación, obteniendo los resultados que se muestran en la Figura 4.11.

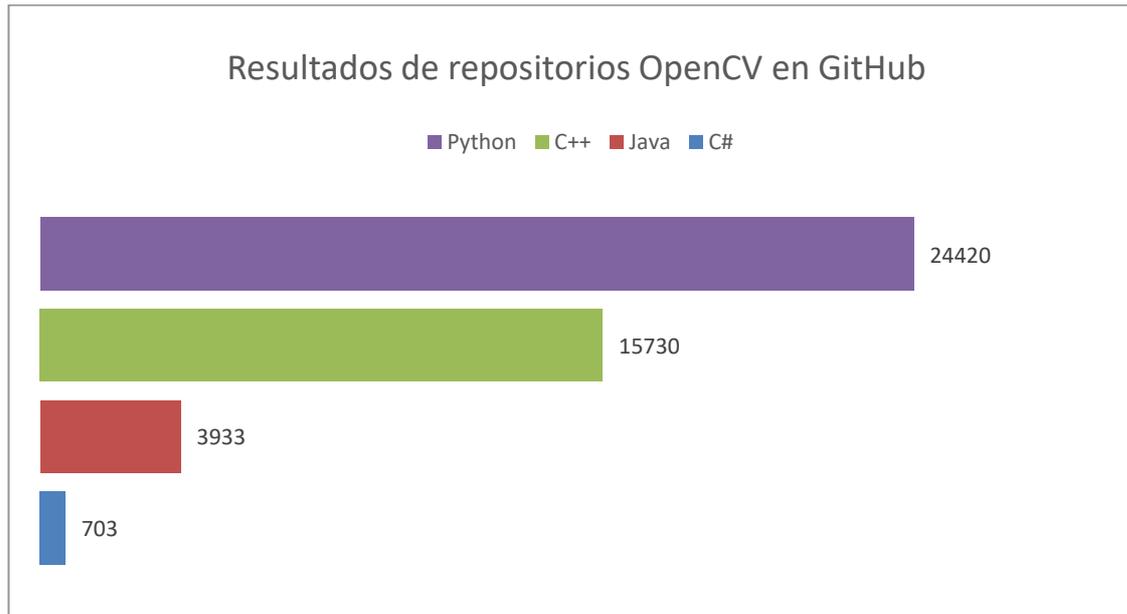


Figura 4.11. Comparación repositorios OpenCV para diferentes lenguajes de programación en GitHub.  
[Fuente: Elaboración propia]

### 4.3.2 OpenCV

Se pueden encontrar una gran cantidad de librerías para el tratamiento y análisis de imágenes, aunque entre todas ellas destaca muy por encima OpenCV.

OpenCV (*Open Source Computer Vision*) como su nombre indica, es una librería de código abierto dirigida principalmente al desarrollo de aplicaciones de visión artificial en tiempo real. Contiene cientos de algoritmos de alto nivel, frecuentemente usados en áreas como reconocimiento de objetos, visión robótica, aprendizaje automático, redes neuronales, etc. Estos llevan actualizándose desde su comienzo en 1999, siendo la versión 4.3.0 la última lanzada hasta el momento, incluyendo mejoras en funciones que se usan en este proyecto.

Son numerosas las ventajas que aporta trabajar con OpenCV, las cuales destacan:

- Soporte multiplataforma: para los sistemas operativos GNU/Linux, Windows, Mac OS y Android, y para arquitecturas de x86, x64 y ARM.
- Compatibilidad con lenguajes de programación populares como Java, MATLAB y Python. Siendo este último el elegido para este proyecto.
- Documentación activa, con ejemplos y tutoriales para favorecer el aprendizaje independiente.

Su estructura es modular, lo que significa que el paquete completo contiene varias librerías compartidas. Los módulos usados en este proyecto son:

- Procesamiento de imágenes (*Image processing*): incluye métodos morfológicos, binarización, tratamiento de histogramas, búsqueda de contornos, etc. Estos se usarán para adaptar y analizar ciertas características de la imagen.
- Características 2D (Features2d): usado para la extracción de características locales con ayuda de descriptores conocidos como ORB, AKAZE, SIFT, SURF, etc. Relacionado con la clasificación de objetos que se llevará a cabo posteriormente.

La versión de OpenCV usada es 4.3.0 compilada con el programa CMAKE para añadir las funciones de los algoritmos SIFT y SURF, ya que estos se deben incluir de manera opcional por el programador.

# 5 DESARROLLO DE LA APLICACIÓN

---

En este capítulo se explicará la solución desarrollada con los métodos explicados en el capítulo anterior. Se abordará el diseño del entorno físico referido a las estanterías y las cajas, y el diseño de códigos generados. Así pues, se verá la conexión y comunicación entre los algoritmos y seguidamente se estudiará cada uno por separado.

La aplicación desarrollada trata de una interfaz gráfica, que permite detectar los materiales que hay en cada caja de la estantería, mostrando visualmente con texto y colores la ausencia o presencia de estos. Así pues, cuando el operario introduzca una caja, el sistema actualizará de forma automática la información sin tener que hacer un procedimiento manual.

Se ha de tener en cuenta que las imágenes de los siguientes capítulos han sido editadas para respetar la confidencialidad de la empresa.

## 5.1 Entorno físico

Actualmente, el entorno físico no es el adecuado para aplicar un sistema de visión. Los problemas principales se encuentran en las estanterías ya que no contienen el suficiente espacio para el uso de cámaras y no tienen suficiente iluminación, y en las cajas ya que estas tienen un diseño cerrado con cada material envuelto en una bolsa de plástico individual.

Se requiere el diseño de un nuevo entorno que acepte las soluciones de visión propuestas, así como otras tecnologías aplicables como el RFID.

### 5.1.1 Estantería

Los requisitos fundamentales se recogen en la siguiente lista:

- El tamaño de la estantería debe ser el adecuado para permitir al operario alcanzar cualquier caja sin dificultad.
- Las cajas deben ser visibles en su totalidad por una o varias cámaras situadas encima de estas.
- Cada estantería debe tener un pequeño visualizador para conocer la información de salida del sistema.
- Se debe incluir la posibilidad de incluir una antena RFID.

Se propone una estantería compuesta de 4 filas con capacidad para 4 cajas cada una de ellas. En la parte superior del hueco de cada caja se instalaría una cámara gran angular para analizarlas de forma individual. Junto a cada cámara se encontraría una fuente de luz que se activa de forma automática según las necesidades de la cámara.

Un pequeño portátil de tipo “*netbook*” acompañaría a cada estantería para ejecutar el programa de visión y recoger la información de cada cámara. Estos equipos de bajo costo son suficientes ya que no se necesitan muestreos y análisis rápidos de las cámaras. Además, estos portátiles permitirían la ejecución de otros programas corporativos de la empresa que pueden resultar muy útiles en conjunto con el programa de visión desarrollado.

Un diseño de la estantería puede ser el que se ve en las siguientes figuras:

En la Figura 5.1 se aprecia que cada fila tiene 4 cámaras, cada una con una fuente de luz que permite iluminar el interior de la caja cuando sea necesario. Además, puede verse el portátil sobre una bandeja que se recorre con un par de guías.

En la Figura 5.2 se puede ver una imagen más amplia de la estantería. En la parte inferior en rojo se representa la zona donde se encontrarían todos los cables, instalaciones eléctricas, antenas RFID, etc.

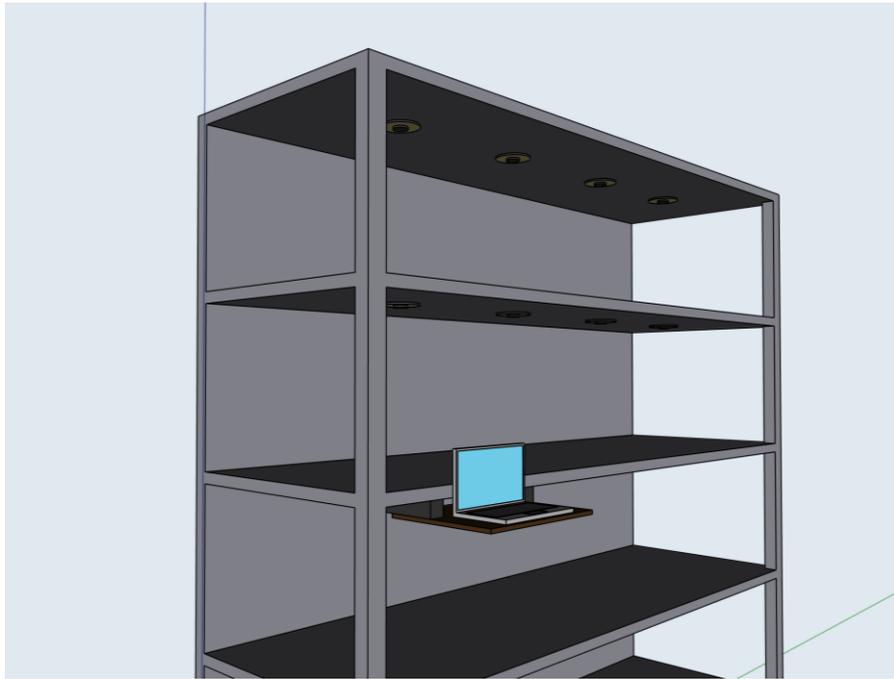


Figura 5.1. Vista 1, diseño propio de la estantería para visión artificial. [Fuente: Elaboración propia]

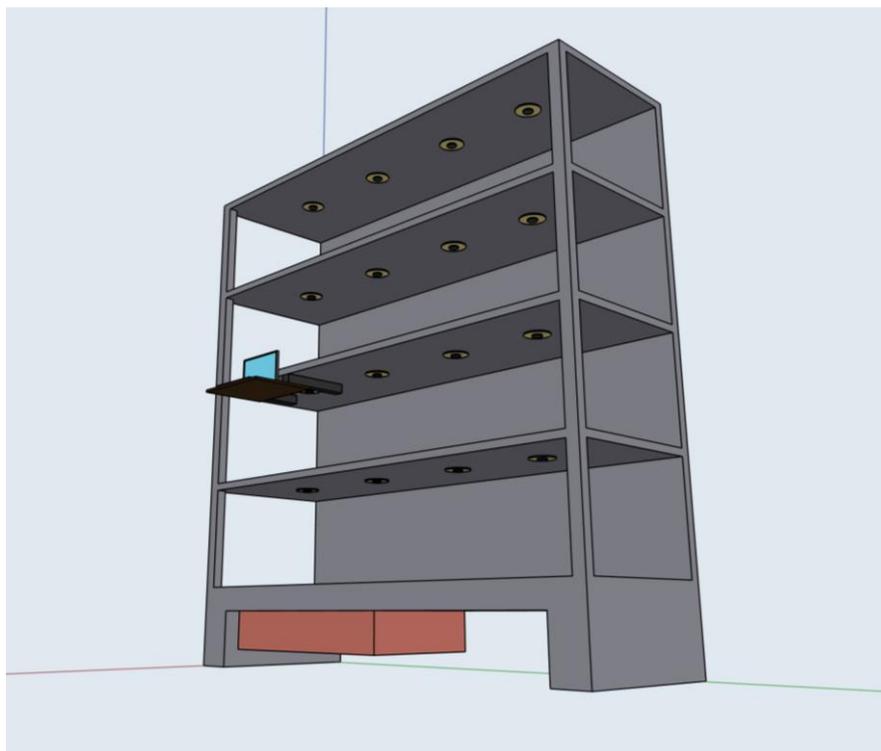


Figura 5.2. Vista 2, diseño propio de la estantería para visión artificial. [Fuente: Elaboración propia]

Esto es un diseño inicial básico de la estantería, que contiene la tecnología suficiente para la aplicación del sistema de visión. También se pueden incluir sensores de luminosidad, presencia, entre otros, que proporcione más información para una programación más adecuada de la solución.

### 5.1.2 Cajas

Se deja de lado las cajas cerradas con muchos materiales en su interior y sin ninguna disposición definida. Las cajas diseñadas para este problema son abiertas, con una espuma roja y silueteadas con un fondo verde, como se aprecia en la Figura 5.4.

Observando la Figura 5.3 se puede ver que incluyen 3 agujeros en las esquinas de la espuma que permite conocer la localización y rotación de las cajas. En la esquina superior derecha se incluye un código QR que al ser leído extrae la información de la referencia de la caja y de todos los materiales que contiene, así como todos los datos que resulten interesantes de guardar.

Este tipo de cajas ya es usado en algunas estaciones de las instalaciones con algunas variaciones, como que la espuma es de color gris haciendo más complicado el procedimiento de binarización por colores.

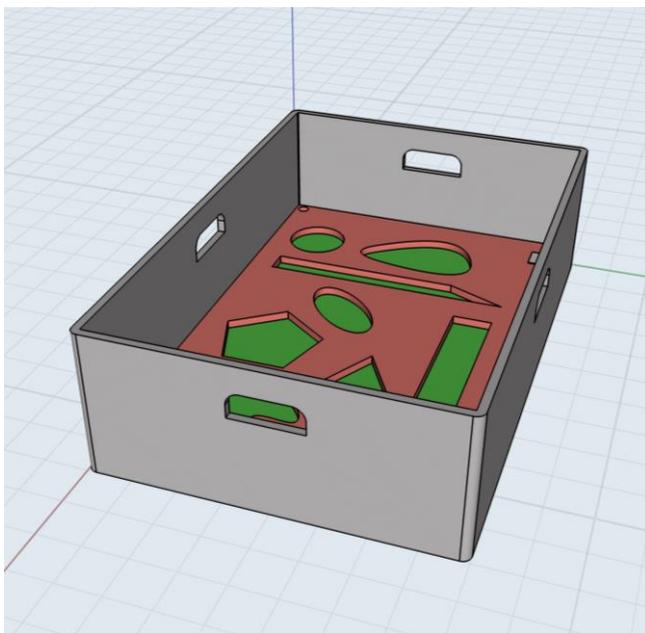


Figura 5.4 Vista 1, diseño propio de las cajas para visión artificial. [Fuente: Elaboración propia]

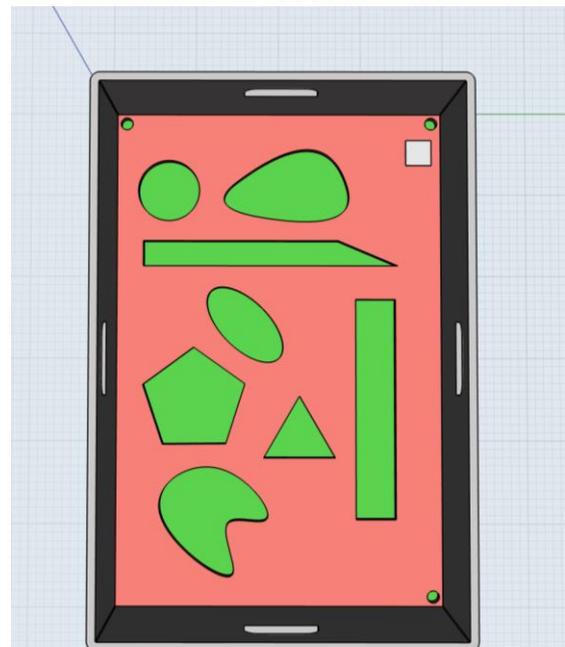


Figura 5.3 Vista 2, diseño propio de las cajas para visión artificial. [Fuente: Elaboración propia]

### 5.1.3 Cámaras

Las cámaras deben tener una conexión tipo “USB” de manera que se puedan conectar al portátil donde se ejecutará la aplicación de visión. Esta característica permite que si alguna cámara deja de funcionar pueda ser reemplazada por otra simplemente conectando el USB en el lugar que se indique. Estas cámaras deben tener una lente de tipo gran angular para ser capaces de captar la caja entera sin necesitar de una gran distancia a la caja.

## 5.2 Estructura de los algoritmos

La solución se compone de 3 códigos fundamentales que son *InterfazGrafica.py*, *Principal.py* y *Funciones.py*. El código *Principal.py* es el encargado de gestionar la unión de los otros dos códigos, dibujando los elementos de la interfaz gráfica con ayuda de *InterfazGrafica.py* y añadiéndole funcionalidad de visión artificial con *Funciones.py*, como se puede ver en la Figura 5.5.

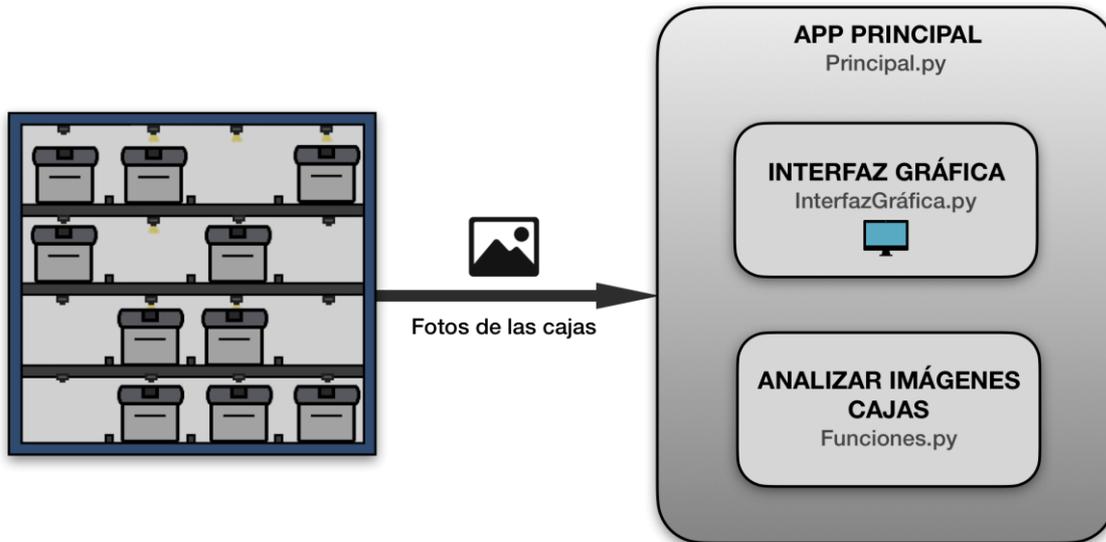


Figura 5.5. Estructura de los algoritmos. [Fuente: Elaboración propia]

### 5.3 Interfaz gráfica: *InterfazGrafica.py*

Es el código que contiene todos los elementos para ser mostrados por la interfaz de la aplicación. La base se ha realizado con la ayuda del programa “*Designer*” que viene incorporado con el módulo *pyqt5-tools*, complementario a *pyqt5*.

#### 5.3.1 Diseño de la aplicación

*Designer* es un editor gráfico que se basa en un sistema de artilugios (conocidos comúnmente como *widgets*) de “arrastrar y soltar”, es decir, que al ser puestos en la ventana generan clases y objetos en el código. Además, se pueden realizar modificaciones en todas las propiedades de cada elemento y se puede ver la previsualización de la aplicación desde el mismo programa, por lo que da una gran flexibilidad a la hora de diseñar. Al guardar el trabajo, el archivo que se genera es del tipo “.*ui*” y para su edición y ejecución en Python tiene que ser transformado a “.*py*”. Esto se realiza desde el terminal con el siguiente comando:

```
pyuic5 -x nombrearchivo.ui -o nombrearchivo.py
```

Este archivo se recomienda no ser modificado, a pesar de ello en este caso se ha optado por hacerlo para mejorar algunos aspectos como la creación de algunas clases para la inserción y eliminación de elementos sin tener que definir cada uno por separado, así como para reducir el código significativamente y mejorar la claridad de lectura, algo muy necesario para poder continuar el proyecto en un futuro si se desea.

El diseño de la aplicación se puede ver en la Figura 5.6 y Figura 5.7 y los elementos que muestra se pueden dividir en dos grupos:

- **El panel izquierdo.** Muestra visualmente un dibujo de la estantería que estamos analizando y su identificación por medio del número de estación donde se encuentra y el número de estantería dentro de dicha estación. Sobre esta irán apareciendo dibujos de cajas en los huecos correspondientes según los operarios las vayan poniendo. Pueden ser de 4 colores distintos indicando así información rápida al operario que la vea. El código de colores se define en la Tabla 5.1. Además, al pulsar sobre ellas aparece la información que contiene en el panel de la derecha.

Tabla 5.1. Significado de los colores de las cajas en la aplicación

Tipos de cajas	Información que aporta
	La caja se encuentra en dicha posición pero que su interior no puede ser analizado debido a algún error.
	La caja contiene todos los materiales.
	La caja contiene algunos materiales, pero no todos.
	La caja no contiene ningún material.

- **El panel derecho.** Se compone de un espacio de gestión de ventanas que sirve para mostrar información acerca de la caja que ha sido seleccionada. El usuario puede elegir entre dos pestañas, la primera muestra la información en forma de etiquetas, con una foto del material (si está disponible) y su referencia. Pueden ser rojas indicando ausencia, o verdes, indicando que se han encontrado correctamente. La segunda pestaña muestra la imagen real del interior de la caja que ha sido tomada por la cámara. Está imagen, tras ser analizada, se le han añadido unos recuadros rodeando a cada material, indicando nuevamente en verde la disponibilidad y en rojo la falta.

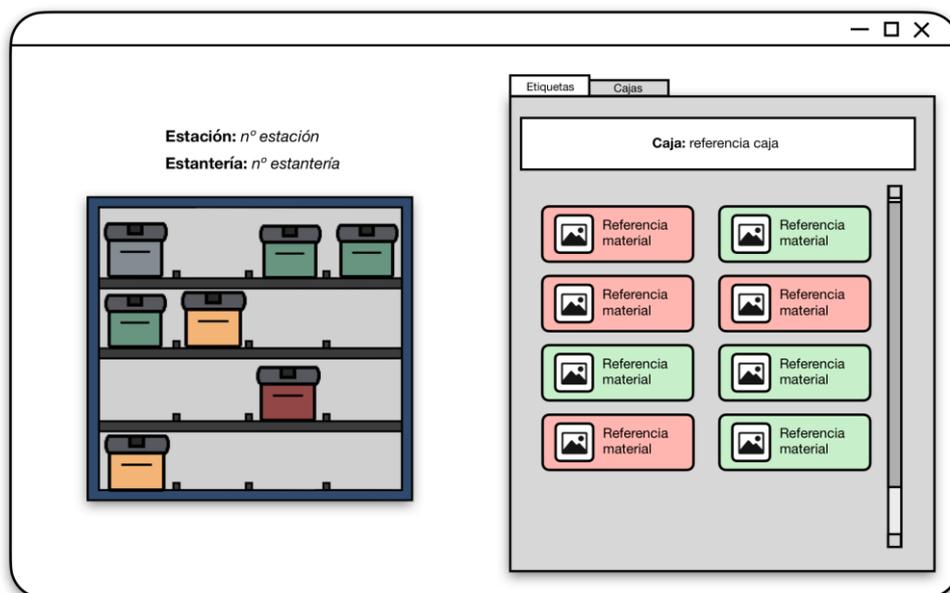


Figura 5.6. Diseño aplicación, pestaña etiquetas. [Fuente: Elaboración propia]

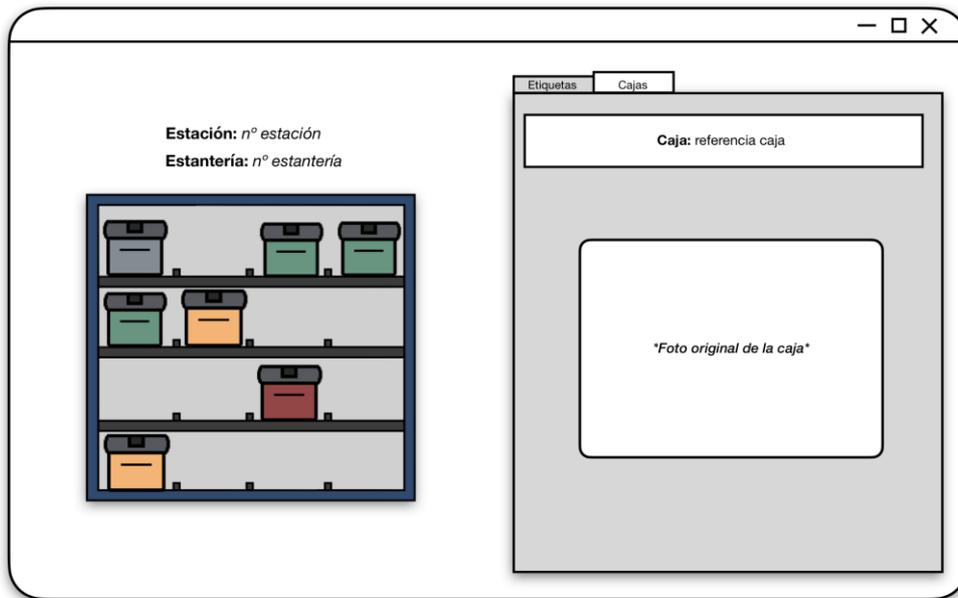


Figura 5.7. Diseño aplicación, pestaña cajas. [Fuente: Elaboración propia]

### 5.3.2 Algoritmo de la aplicación

Una vez realizado el diseño previo de la aplicación en el programa “*Designer*”, se pasa a un formato legible por Python. Se ha de tener en cuenta que, al tratarse de un programa para la ayuda del diseño, en el código se han añadido cada elemento por separado y de forma poco intuitiva. Por este motivo, se ha modificado para mejorar la claridad y reducir considerablemente su tamaño.

Cada widget introducido se traduce en un objeto de una clase, ya definida en la librería *Pyqt5*, como son etiquetas de texto, botones, espaciadores, etc. Sin embargo, la aplicación necesita un número indeterminado de etiquetas a mostrar, por lo que se ha creado una clase para este fin, llamada “*Etiqueta\_material*”, que permite generar y eliminar etiquetas desde el código “*Principal.py*”.

Se ha de tener en cuenta que la interfaz gráfica no tiene funcionalidad de no ser por los siguientes códigos explicados a continuación.

## 5.4 Funciones de visión: *Funciones.py*

El código “*Funciones.py*” es el que añade los algoritmos de visión artificial a la aplicación. Se compone principalmente de 2 funciones, “*AnalizaCaja*” y “*BuscaMateriales*”. La primera es llamada a través del código “*Principal.py*” para realizar el análisis de las imágenes tomadas. De esta forma, realiza un procesamiento para encontrar la caja y luego se ayuda de la segunda función para detectar los materiales. Los pasos realizados para realizar dicho análisis de la imagen son los siguientes:

- **Búsqueda de puntos de referencia.** Cada cámara está en constante funcionamiento. Cuando una caja es colocada, la forma de saber de su presencia es con la ayuda de los puntos de referencia que presenta que se pueden observar en la Figura 5.8. Además, estos puntos deben servir también para saber la posición y orientación de esta. Como el fondo de la caja es rojo, se ha decidido que estos puntos sean huecos circulares en las esquinas de la espuma, realizados en el proceso de silueteados. Sin embargo, no solo no hacen falta puntos en cada esquina, sino que tener 4 puntos de referencia complica la tarea de encontrar la orientación correcta de la caja. Por eso, se ha decidido estandarizar el proceso omitiendo el punto de la esquina inferior izquierda. A continuación, para encontrar dichos puntos, cada imagen tomada por la cámara se pasa al espacio de color “*HSV*”. Posteriormente, se crea una máscara solamente

con el color rojo de la imagen, mostrada en la Figura 5.9. Luego, para mejorar la discretización, se procesa mediante el método morfológico conocido como “*closing*” y se aplica una pequeña distorsión. Llegados a este punto, se aplica el método “*HoughCircles*” para detectar círculos en la imagen, obteniendo de esta forma su localización en los ejes coordenados, así como el radio de circunferencia. Finalmente, entre todos los posibles puntos encontrados, se pueden hallar cuales son los que corresponde a cada esquina por sabiendo que son los círculos más externos de la imagen.

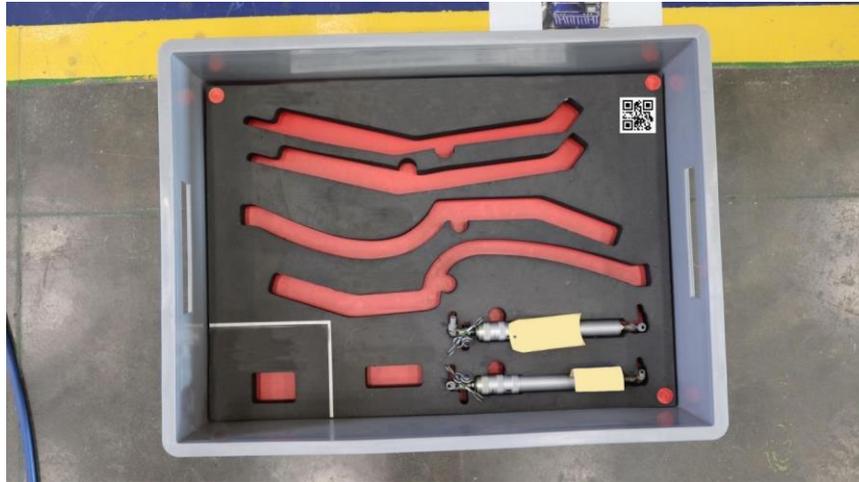


Figura 5.8. Foto de la caja con puntos de referencia. [Fuente: Elaboración propia]

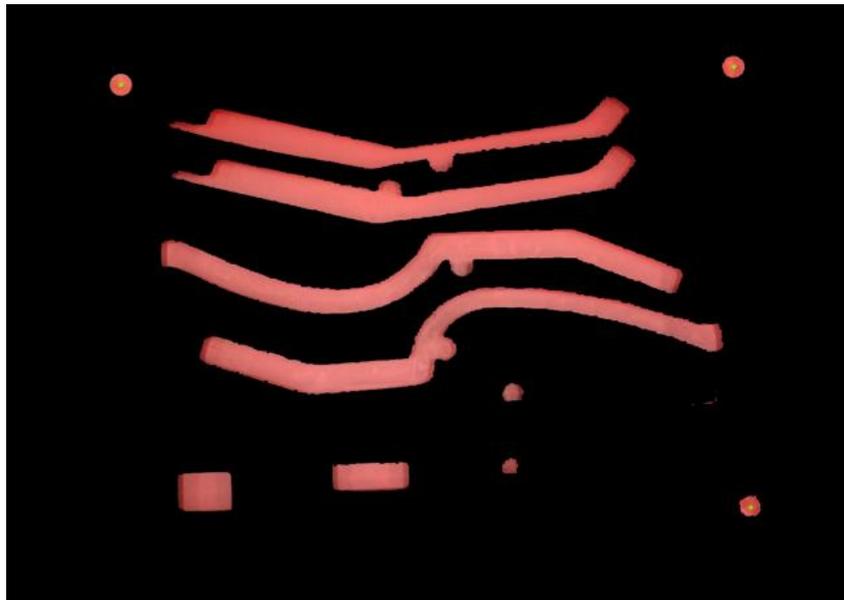


Figura 5.9. Caja discretizada por color rojo. [Fuente: Elaboración propia]

- **Recorte de la imagen.** El cuarto punto de la esquina que falta se puede reconstruir a partir de los otros 3. Este último, es necesario para poder aplicar métodos de cambio de perspectiva. De esta forma, se puede centrar la parte del interior de la caja y se puede definir un tamaño estándar de imagen de 600x400. Ya que es necesario cuando se busquen los materiales hablar de las mismas posiciones y medidas tanto de la imagen original como de la de referencia. Si este primer paso no sale bien, salta un error que nos informa que no se ha encontrado la caja o no es posible centrarla.

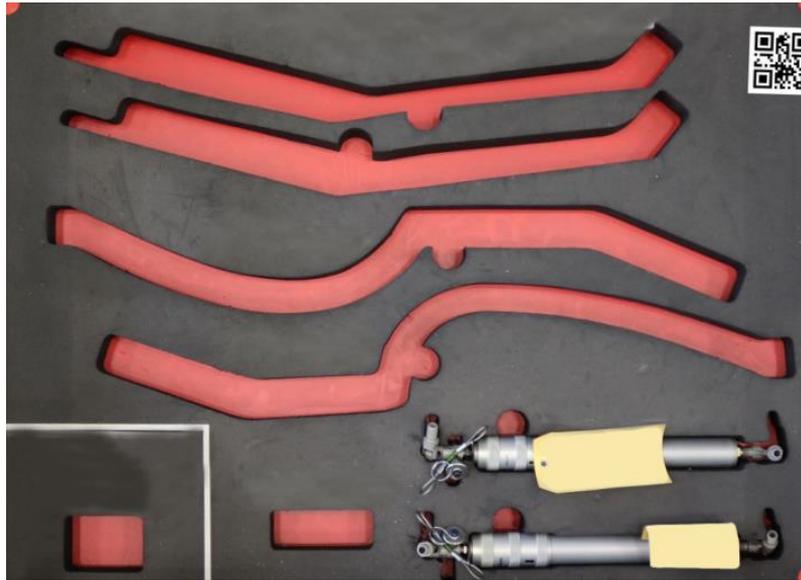


Figura 5.10. Caja recortada por puntos de referencia. [Fuente: Elaboración propia]

- **Extracción de información mediante QR.** Cada caja viene acompañada de un código QR que, tras haber rotado la imagen con la orientación adecuada, se encuentra siempre en la esquina superior derecha. Definir la localización del código en esta esquina no es totalmente necesario, pero acelera el proceso si solo analizamos esta parte de la imagen y no entera. Es por ello por lo que, sabiendo que la imagen mide 600x400, recortamos un cuadrado de 150x150 debajo del punto de referencia de la esquina correspondiente, como puede apreciarse en la Figura 5.11. Gracias al módulo de *pyzbar*, se puede analizar este cuadrado y sacar la información del QR.

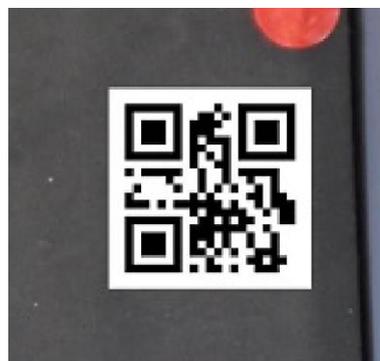


Figura 5.11. Recorte de la caja para analizar código QR. [Fuente: Elaboración propia]

- **Búsqueda de la caja de referencia en base de datos.** Una vez extraída la referencia de la caja mediante el QR, se busca en la base de datos del programa, como indica la Figura 5.12. Esta imagen de referencia se trata del diseño realizado previamente de cada caja, por el equipo de “*LEAN Manufacturing*” de la empresa, que puede observarse en la Figura 5.13. Este diseño, necesario para el recorte de la espuma interior, sirve de ayuda en esta aplicación a la hora de encontrar los materiales, ya que así se sabe de antemano donde deben estar situados. Una vez encontrada esta imagen de referencia, pasa a realizarse el procedimiento de búsqueda de materiales llamando a la función “*BuscaMateriales*”, que devuelve como resultado 3 variables. Un vector “materiales” que indica con 0 los que no han sido encontrados y con 1 los que sí, una variable de tipo entero “falta” que indica la cantidad de ceros encontrados en el vector anterior, y una imagen que es una copia de la tomada por la cámara, recuadrando con rectángulos a cada material, indicando en verde que se encuentran y en rojo que faltan. Finalmente, se guarda esta imagen en la base de datos para ser mostrada posteriormente por la interfaz de usuario. Si alguna parte

de este proceso sale mal, salta un error indicando que no se ha podido extraer correctamente la información de la caja.



Figura 5.12. Proceso de búsqueda de caja de referencia en base de datos. [Fuente: Elaboración propia]

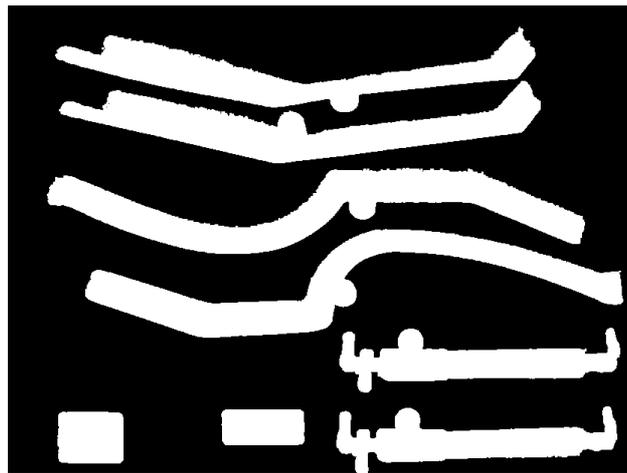


Figura 5.13. Máscara binarizada de la caja de referencia. [Fuente: Elaboración propia]

- **Búsqueda de materiales.** Se trata de segunda función principal del código que recibe como entradas la imagen de referencia sacada de la base de datos y la imagen tomada por la cámara. Como salida la información que devuelve es la indicada en el paso anterior. Dentro de esta función se realiza un primer procesamiento de imágenes que trata de discriminar la imagen de referencia por el color rojo, obteniendo así una máscara binarizada que se mejora mediante métodos morfológicos. Recordando, la imagen de referencia contiene el diseño del interior de la caja, por lo que gracias a esto queda como resultado una máscara que contiene únicamente la forma de los huecos donde luego van los materiales. La máscara se redimensiona con las medidas 600x400 que tiene la imagen tomada por la cámara después de ser recortada. Luego, se aplica la función “*findContours*” para encontrar la posición de los huecos y el rectángulo que los contiene. Llegados a este punto, ya se sabe la posición de los materiales por lo que pueden ser analizados uno a uno en la imagen tomada por la cámara. Para añadir robustez a la detección de estos materiales, a partir de este punto se aplicarán dos métodos diferentes, un método usando procesamiento de imágenes y otro método usando extracción de características 2D.
  - o **Método por procesamiento de imágenes.**

La búsqueda de materiales por procesamiento de imágenes se ha realizado comparando los píxeles rojos del hueco en la imagen de referencia y la imagen original, de esta manera se sabe si hay un objeto colocado que tapa el fondo rojo de la caja.

Cuando se discretizan estas imágenes por color, hay dos formas de analizar los resultados. Una de ellas es sumando todos los píxeles resultantes de la imagen original de tal forma que si hay más de un 30% de puntos rojos de referencia quiere decir que no hay objeto ocultando este fondo. Este porcentaje puede ser ajustado según los objetos que contiene la caja para unos mejores resultados.

La segunda forma es haciendo una búsqueda de contornos de la binarización resultante y hacer la misma comparación anterior con el cálculo del área del elemento con mayor contorno. Se ha

decidido utilizar este segundo método debido a que suele ocurrir que cuando se recorta el rectángulo que contiene al material, este puede contener también partes de otros, por lo que puede que algunos píxeles rojos encontrados sean de otro material que no es el que estamos analizando. Estos píxeles son pocos, pero pueden activar la detección de un material cuando no lo hay.

En la Figura 5.14 se observan los resultados de este método cuando se encuentran o no materiales en la caja.

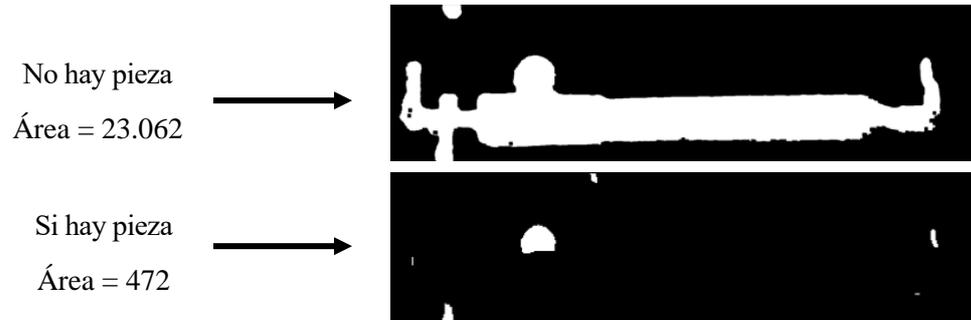


Figura 5.14. Resultados búsqueda de materiales por conteo de píxeles rojos.  
[Fuente: Elaboración propia]

El pseudocódigo de este procedimiento es el siguiente:

```

Discretizar imagen_referencia por el color rojo
Discretizar imagen_original por el color rojo
Para los contornos de la imagen de referencia
    Calcular área del contorno
    Recortar rectángulo del material en la imagen original
    Para los contornos en el recorte de la imagen original
        Hallar el área_máxima encontrada
        Si área_máxima encontrada < (coeficiente similitud) * área_contorno
            Añadir 1 al vector materiales (indicando que se ha encontrado)
        Sino
            Añadir 0 al vector de materiales indicando que no se ha encontrado
        Fin
    Fin
Fin

```

Esta forma de detección únicamente sirve para saber si hay un objeto, pero no para saber si es el que corresponde en esa posición. La ventaja es que es rápido y como la empresa sigue la filosofía 5S, solo con este método basta para obtener una información correcta en la mayoría de los casos. A pesar de esto, no siempre se coloca el material en el hueco correspondiente, por lo que tenemos un segundo método comparando las características del material con una imagen de referencia y la original.

- **Método de extracción de características locales.**

Este método es complementario al anterior y se usa únicamente en el caso de que haya una pieza en el hueco analizado, ya que en el caso contrario el procedimiento anterior es suficientemente robusto.

Como la imagen del diseño de la caja indica la posición de cada hueco, se puede hacer un análisis individual para evitar fallos de emparejamiento. Para identificar el material se necesitan imágenes de referencia tomadas de la base de datos.

De esta forma, se extraen las características de ambas imágenes con los métodos SIFT o SURF, y se escogen los puntos más parecidos entre ellas. Esta unión de puntos se realiza a través del método FLANN, que contiene varios algoritmos optimizados para ello. Así pues, el número emparejamientos buenos encontrados se compara con el número típico que suele dar esta identificación en un caso positivo.

Si no supera un umbral, se dice que el material no es el correspondiente y de lo contrario se valida la identificación.

El pseudocódigo de este procedimiento es el siguiente:

```
Detección de características en la imagen_referencia
Detección de características en la imagen_original
Unión de características con FLANN
Descarte de los peores emparejamientos
Si el número de emparejamientos buenos es razonable
| Se recuadra la posición de la pieza
Sino
| Se indica que no es la pieza correspondiente
```

Los resultados que se obtienen de este procedimiento se pueden ver en la Figura 5.15.

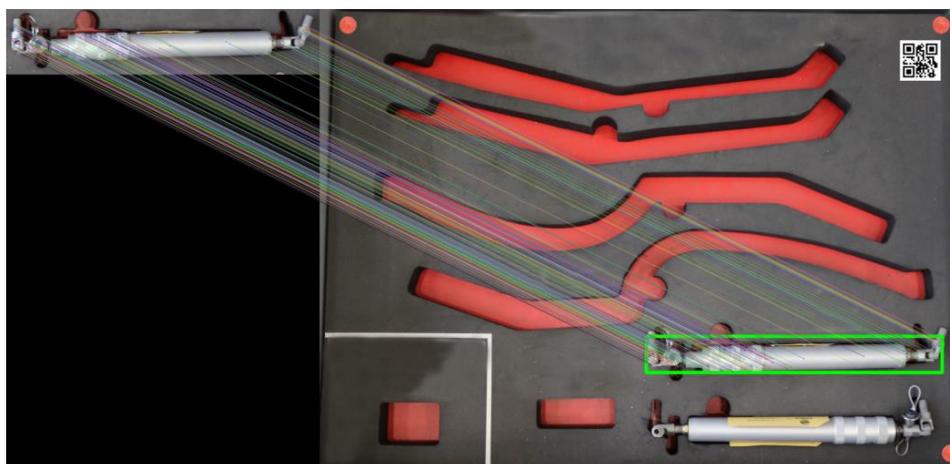


Figura 5.15. Resultado de identificación de materiales por extracción de características locales.

[Fuente: Elaboración propia]

- **Comprobación de igualdad.** Normalmente una caja está en la estantería por un periodo de tiempo largo, y en cada iteración del programa, como se verá posteriormente, al botón de la caja de la interfaz se le asigna una acción. En cada ciclo, el programa asigna de nuevo la misma acción al mismo botón, ralentizando de esta manera la interfaz, haciendo imposible la interacción con ella. Por esta razón, antes acabar el análisis de la imagen, se comprueba si la caja es la misma y además contiene los mismos materiales. De ser así se ignora la asignación, y si no, se elimina la conexión anterior y se asigna de nuevo con la nueva caja o distribución de materiales. De esta manera siempre se tiene una acción solo una vez asignada a un mismo botón y la interfaz se mantiene fluida.

## 5.5 Conexión de la interfaz con la funcionalidad: *Principal.py*

El código “*Principal.py*” sirve para la unión de la interfaz gráfica con las funcionalidades de visión. Para realizar esta tarea primero inicializa los elementos gráficos y luego ejecuta en paralelo un bucle infinito para analizar las imágenes que obtienen las cámaras. Es este bucle el que, según los materiales detectados, une los botones de la interfaz con una acción específica, mostrar una serie de datos por pantalla. Esto lo hace por medio de señales, lo que permite que la aplicación esté activa todo el rato y cuando se pulse sobre el botón de la caja se realice la acción correspondiente sin interrumpir el flujo del programa.

Lo que hace que se ejecuten estos pasos es la inicialización de una clase llamada “*Aplicación*”, y al proceso paralelo que llama se denomina “*AnalisisCamaras*”. Este último es el que se mantiene ejecutándose sin interacción del usuario, mientras que las funciones definidas en la clase “*Aplicación*” son llamadas cuando el operario hace uso del programa. A continuación, se explican estas dos en detalle:

- **Aplicación.** Se encarga de administrar los elementos de la interfaz gráfica definidos en “*InterfazGrafica.py*” con la ayuda de varias funciones. Además, estas son llamadas por el proceso paralelo o por el usuario, y son:
  - **Init:** es la función que ejecuta el código principal al inicializar la clase. Su finalidad es mostrar los elementos iniciales y ejecutar en segundo plano el proceso de “*AnalisisCajas*”.
  - **PonerCaja:** muestra una caja del color adecuado en el hueco correspondiente de la estantería, a la vez que conecta dicho botón a la acción “*MuestraCaja*”. Esta función es llamada por el proceso según vaya analizando las cajas correspondientes.
  - **QuitarCaja:** quita la caja visualmente de la interfaz cuando se retira esta. Únicamente es llamada por el proceso paralelo.
  - **MuestraCaja:** sirve para mostrar, en el panel derecho de la interfaz, la información de las cajas, tanto en forma de etiquetas como en forma de imagen. Esta función se ejecuta cada vez se pulsa sobre algún botón de caja. Para ello utiliza la información sacada al analizar la caja. También crea las etiquetas con ayuda de la clase “*Etiqueta\_material*” vista en el apartado 5.2.2 de este documento. Es ejecutada cuando el usuario pulsa sobre algún botón de caja de la interfaz.
  - **EliminaCaja:** vacía la información del panel derecho de la interfaz. La utiliza “*MuestraCaja*” para limpiar la información que había anteriormente.
- **Materialescámara:** es el proceso paralelo que analiza las imágenes y, según los resultados actúa sobre la interfaz. Es un bucle infinito que sigue los siguientes pasos:
  - **Toma de imágenes.** Cada cámara está conectada a un puerto del ordenador y se identifica individualmente. Esta toma de imágenes se realiza individualmente en cada iteración del bucle.
  - **Análisis de imágenes.** Se ejecuta la función “*AnalizaCaja*” del código “*Funciones.py*”. Recibe la imagen actual y la información de la caja que había anteriormente en esa posición. De esta forma obtiene los datos correspondientes a la caja actual, junto con la señal de error para tratar excepciones.

- **Administración de excepciones.** Se ejecutan las acciones correspondientes según la excepción que haya saltado. Pueden ser de dos tipos, el primero es porque falle alguna parte del proceso indicándose con números negativos del -1 al -4, y el segundo son números positivos que indican que el proceso se ha terminado por otra causa. En el caso de ser negativas, primero se muestra el mensaje, luego se desconecta el botón de la caja de la acción y finalmente se elimina la caja. En el caso de ser positivas, no implica fallo en el análisis de la caja por lo que simplemente se pasa a analizar la siguiente saltándose el último paso del proceso. Finalmente, el significado de todas ellas se representa en la Tabla 5.2.
- **Actuación sobre interfaz gráfica.** Con la información de la caja, si no ha habido ninguna excepción, se emiten los datos a la función “*Ponercaja*”. Estos incluyen el número de la caja en la estantería, el vector de los materiales, la referencia de la caja y su color.

Tabla 5.2 Excepciones y su significado.

N.º Excepción	Significado
-1 o 0	La cámara está desconectada ya sea porque no se ha conectado nunca (0) o porque no puede obtener la imagen o se ha desconectado posteriormente (-1).
-2	No se han podido encontrar todos los puntos de referencia de la imagen.
-3	No se ha podido leer el QR o encontrar la información de la caja en la base de datos o ha habido algún problema en la búsqueda de materiales.
1	Se trata de la misma caja por lo que no es necesario conectar de nuevo la caja con la acción.



# 6 RESULTADOS EXPERIMENTALES

En este capítulo se mostrará el resultado final de la aplicación desarrollada, tanto su aspecto físico como su funcionamiento. También se analizarán los resultados obtenidos por los métodos de extracción de características SIFT y SURF, y se comentarán las limitaciones actuales de la solución para dar pie a mejoras futuras del procedimiento. Al igual que en el capítulo anterior, las imágenes han sido editadas para mantener la confidencialidad de la empresa.

## 6.1 Interfaz gráfica y funcionamiento general

El aspecto resultante de la interfaz gráfica se puede observar en todas las figuras de este apartado, donde se aprecian las funcionalidades descritas en el apartado anterior. Cuando se pulsa sobre una de las cajas de la estantería se puede ver su información correspondiente en el panel derecho, ya sea en forma de etiquetas o en forma de vista real de la caja (ver Figura 6.1 y Figura 6.2).

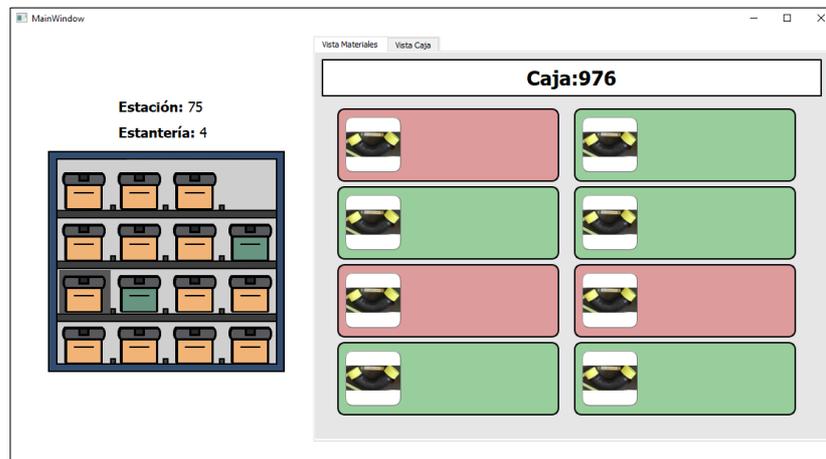


Figura 6.1. Interfaz final: vista etiquetas de caja con algunas piezas. [Fuente: Elaboración propia]

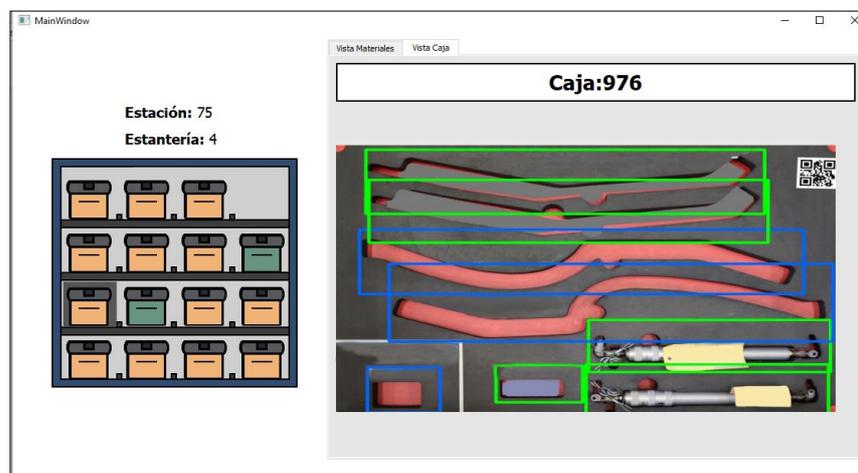


Figura 6.2. Interfaz final: vista imagen real de caja con algunas piezas. [Fuente: Elaboración propia]

En el caso de pulsar sobre una caja verde, se contempla que esta contiene todas las piezas correspondientes, como se puede ver en la Figura 6.3. Por el contrario, si la caja estuviese vacía, no conteniendo ninguna de las piezas, su color pasa a ser rojo, como sucede en la Figura 6.4. Un caso intermedio sería el de las cajas amarillas, las cuales disponen de parte del material.

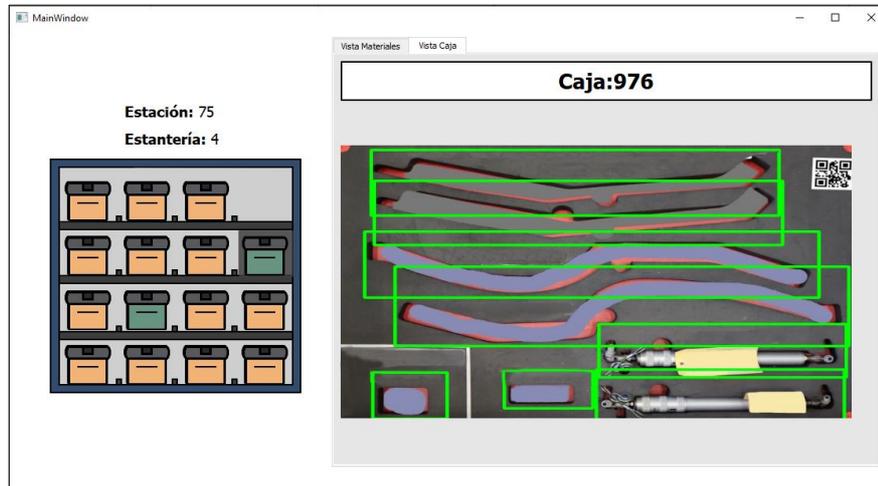


Figura 6.3. Interfaz final: vista imagen real de caja con todas sus piezas. [Fuente: Elaboración propia]

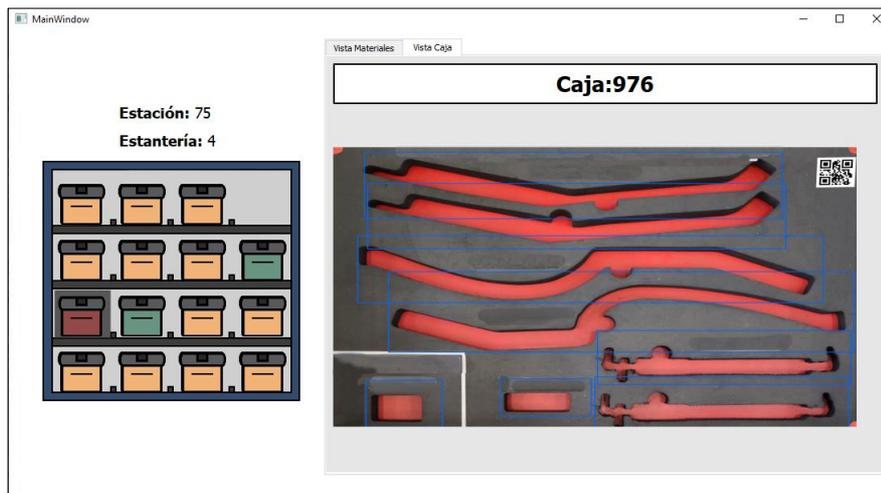


Figura 6.4. Interfaz final: vista imagen real de caja sin piezas. [Fuente: Elaboración propia]

Se puede afirmar que el programa funciona de manera correcta, sin ralentizarse por el análisis continuo de cajas. La detección de los cambios se realiza en un tiempo diferente dependiendo de si se requiere la aplicación de los métodos de extracción de características locales. En cualquier caso, el tiempo que se tarda en analizar todas las cajas es de algunos segundos, lo suficiente para esta aplicación ya que no se esperan muchos cambios del contenido de las cajas durante la jornada, y no se requiere una detección instantánea en tiempo real, por lo que el tiempo de cómputo es admisible.

Para conocer más detalles sobre este proceso, véase apartado 5.

Para confirmar lo anterior, en el siguiente punto se presentan los resultados obtenidos con el uso de los métodos SIFT y SURF.

## 6.2 Resultados de los métodos SIFT y SURF

El empleo de los métodos SIFT y SURF tiene lugar cada vez que se localiza una pieza, permitiendo identificar si se trata del material correspondiente o no, en cada caso. Para ello se extraen los puntos característicos de ambas imágenes (referencia y original), conocidas como *keypoints*, se describen a través de los descriptores, y se determina el número de emparejamientos hallados (*matches*).

Tras eliminar los peores emparejamientos, se decide si la pieza se corresponde con la imagen de referencia. Para ello se hace uso de un umbral típico predeterminado que cada pieza tiene asignado, de forma que establece un número de *matches* mínimo para considerar válida la detección. Por este motivo, es importante tener en cuenta el número de uniones de puntos característicos realizado o número de *matches*.

A continuación, se analizarán los resultados obtenidos en la localización de tres objetos comunes muy diversos en cuanto a sus características (forma geométrica, brillo, color, número de detalles, ...). La detección se realizará mediante imágenes capturadas con distintas condiciones ambientales (con fuente de luz artificial intensa, o sin ella), y haciendo uso de los distintos métodos ya mencionados (SIFT y SURF). Adicionalmente, se diferencia una situación excepcional que podría contemplarse en un caso real: la ocultación parcial entre los objetos.

Las imágenes de referencia usadas pueden verse en la Figura 6.5. En ellas se muestran los 3 objetos empleados en el proceso (gorra, pasaporte y teclado), con fuente de luz artificial intensa (conocida comúnmente como *flash*) en la parte izquierda, o sin ella (derecha).



Figura 6.5. Imágenes de objetos de referencia. [Fuente: Elaboración propia]

Se realiza una mejor detección cuando los objetos tienen una forma rectangular, puesto que se ajusta perfectamente a la matriz que contiene la imagen, buscando únicamente puntos característicos pertenecientes a estos objetos. Este es el caso del pasaporte y el teclado inalámbrico.

Por el contrario, la gorra tiene una forma irregular, como puede ocurrir con la mayoría de piezas de la industria. En este caso, se debe de aplicar una máscara binaria al proceso de búsqueda de puntos característicos de manera que no se extraiga información del fondo, que no es de interés. Esta máscara se obtiene de un proceso de binarización por color usado en el procesado de imágenes anterior.

En la Figura 6.6 se puede apreciar que, dependiendo de las condiciones en la que se toma la imagen, la binarización aporta diferentes resultados. Es destacable que en la máscara obtenida con el uso de *flash*, en este ejemplo en concreto, se pierde información del dibujo de la gorra que puede ser importante para caracterización de este objeto.

### Máscara Gorra con *flash* vs. Máscara Gorra sin *flash*



Figura 6.6. Máscaras para la extracción de características locales (objeto 1: gorra).  
[Fuente: Elaboración propia]

Además, también es posible afirmar que cada objeto se caracteriza con las descripciones de las zonas que rodean a cada punto característico. Así pues, se pretende buscar el método y las condiciones que favorezcan un mayor número de estos. En la Figura 6.7 se comparan los puntos encontrados en las imágenes de referencia con los métodos SIFT y SURF.

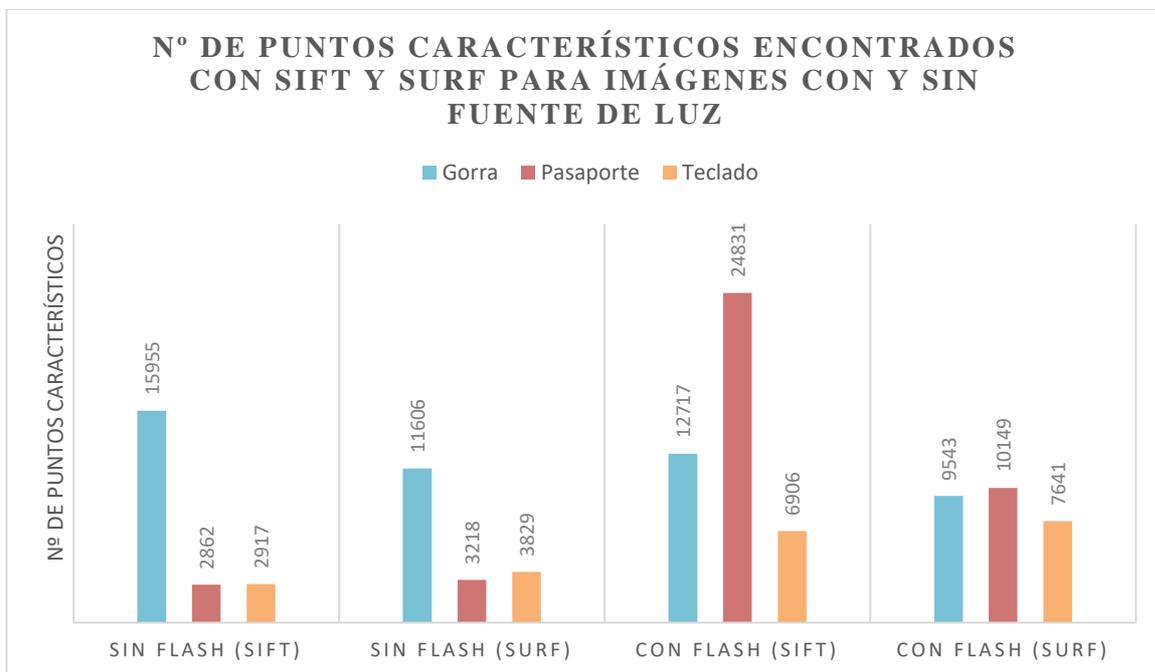


Figura 6.7. N.º de puntos característicos encontrados con SIFT y SURF con y sin fuente de luz externa.  
[Fuente: Elaboración propia]

De esta gráfica se puede deducir que ambos métodos aportan resultados similares. En las imágenes donde no se ha usado fuente de luz artificial, SURF suele encontrar más puntos característicos (pasaporte y teclado) y lo contrario ocurre cuando sí se hace uso de ella, siendo SIFT superior (gorra y pasaporte).

Además, se puede observar que el uso de iluminación externa siempre es favorable para la búsqueda de estos puntos en ambos métodos. Se destacan los resultados obtenidos en el pasaporte, que pasa de los 2.862 puntos encontrados con el método SIFT, a 24.831 tras aplicar una fuente de luz externa. Esta gran diferencia puede explicarse volviendo a la Figura 6.5, donde se aprecia que la imagen del pasaporte con el uso de iluminación externa consigue destacar la rugosidad del material y en caso de no usarla esta se pierde y se homogeniza.

Hay que tener en cuenta que estos resultados sirven únicamente de ejemplo. Para sacar unas conclusiones adecuadas y fiables se deben realizar muchas más comparaciones con un mayor número de objetos variados, poniendo a prueba ambos métodos en diferentes condiciones.

Una vez se caracterizan las imágenes de referencia, también se buscan puntos característicos en las imágenes donde se quieren identificar los diferentes objetos. De esta forma, se pueden comparar los descriptores obtenidos con los de los objetos de referencia con distintos métodos como el de “fuerza bruta” o “FLANN”, y se obtienen los diferentes emparejamientos.

Conviene recordar que un descriptor es un vector multidimensional, y que un emparejamiento se conforma de un par de dichos descriptores que son los más similares entre los demás. Para ello, se debe calcular la similitud entre dos vectores, usando para ello la “*distancia de Hamming*”, para descriptores binarios, o la “*distancia Euclídea*”, para descriptores reales como son SIFT y SURF. Con esta información se puede eliminar los peores emparejamientos para evitar errores y mejorar la identificación.

A continuación, se han llevado a cabo pruebas con las mismas condiciones anteriores, obteniendo los resultados que se muestran en la Figura 6.8.

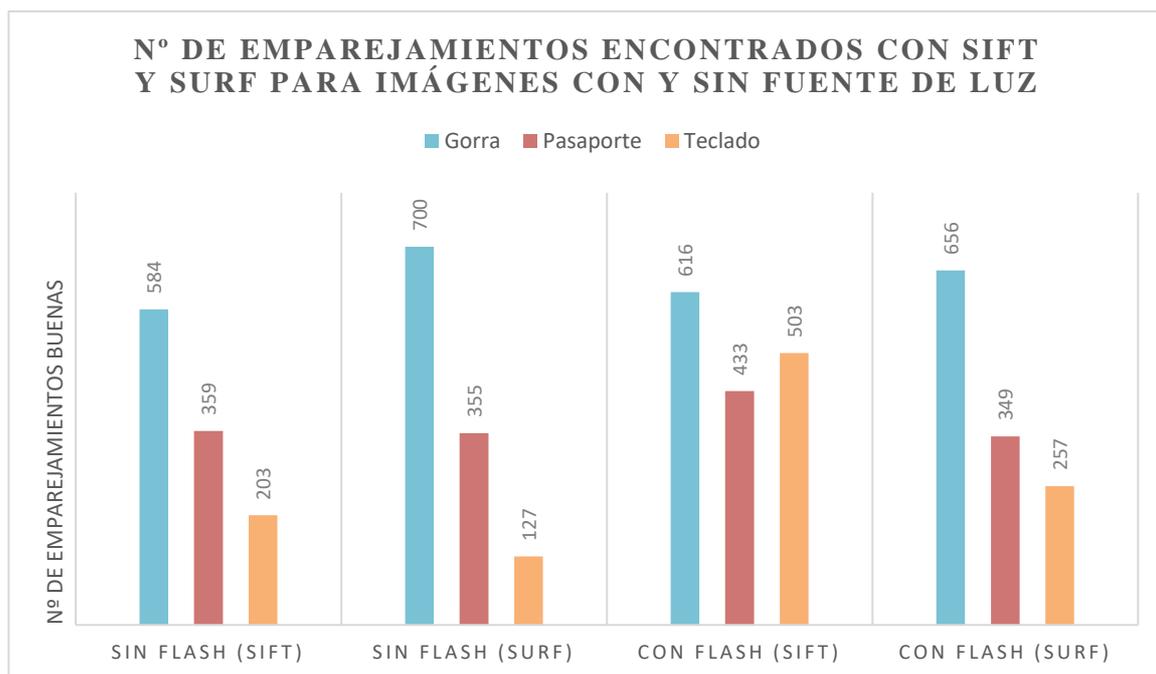


Figura 6.8. Emparejamientos encontrados con SIFT y SURF para imágenes con y sin *flash*.

[Fuente: Elaboración propia]

Se puede comprobar que en la mayoría de los casos el método SIFT consigue hacer más emparejamientos que SURF. Además, en ambos métodos el uso de iluminación artificial supone un gran aumento del número de emparejamientos encontrados, aunque con SIFT el aumento es mucho mayor en todos los objetos.

Tras identificar un objeto, se localiza, tratando de encontrar la orientación de este respecto a la del mismo objeto en la imagen de referencia. Esto se realiza a través de del método RANSAC y se obtienen los siguientes resultados de la Figura 6.9.

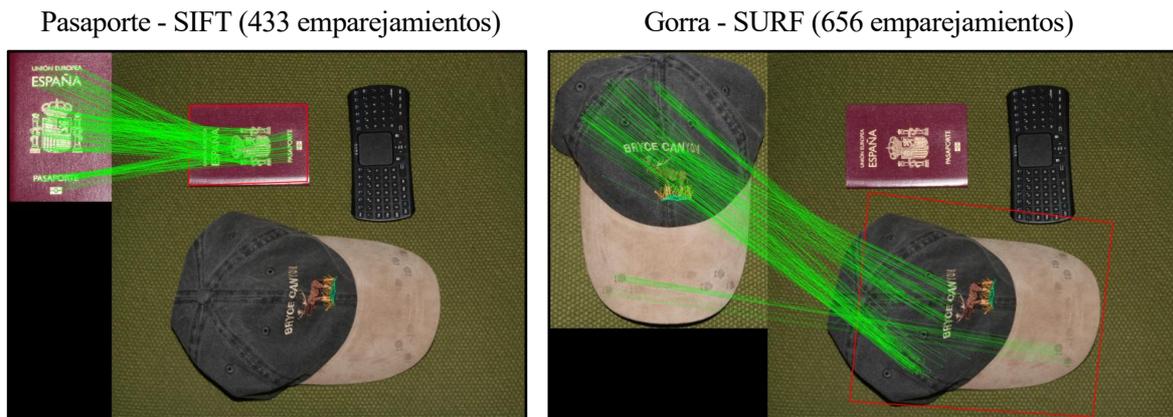


Figura 6.9. Resultados de la identificación de objetos. [Fuente: Elaboración propia]

Hasta este punto, ambos métodos (SIFT y SURF) consiguen dar una buena solución al problema de búsqueda y localización de materiales para diferentes condiciones. No obstante, se puede dar el caso de que los objetos o piezas industriales se encuentren ocultas parcialmente por otras. A continuación, se han realizado las pruebas anteriores con nuevas imágenes que contemplen este caso de ocultación parcial, ver Figura 6.10.

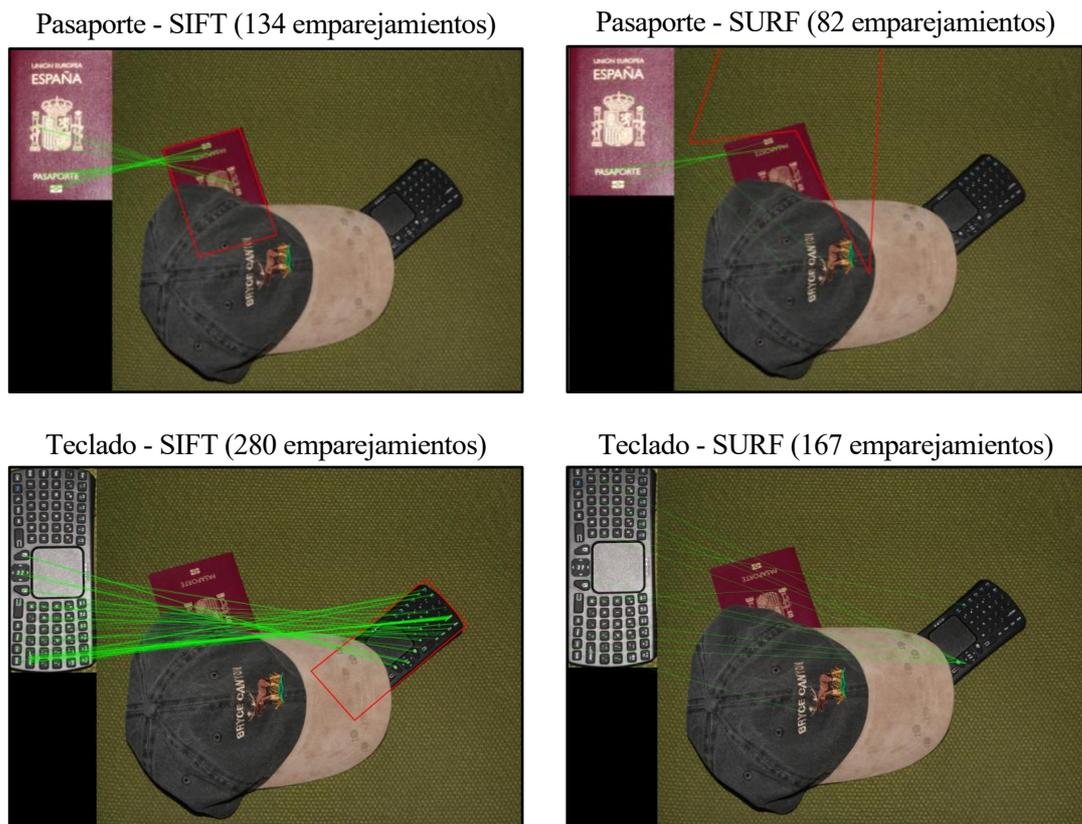


Figura 6.10. Resultados de la identificación de objetos ocultos parcialmente. [Fuente: Elaboración propia]

Se puede comprobar que con una buena iluminación, el método SIFT consigue resolver correctamente el problema de identificación y localización de objetos incluso con la ocultación parcial de estos. Sin embargo, SURF falla en todos estos casos, pues se debe recordar que este es un método simplificado del anterior.

En el caso de que no haya una fuente de luz complementaria los resultados se pueden ver en la Figura 6.11.

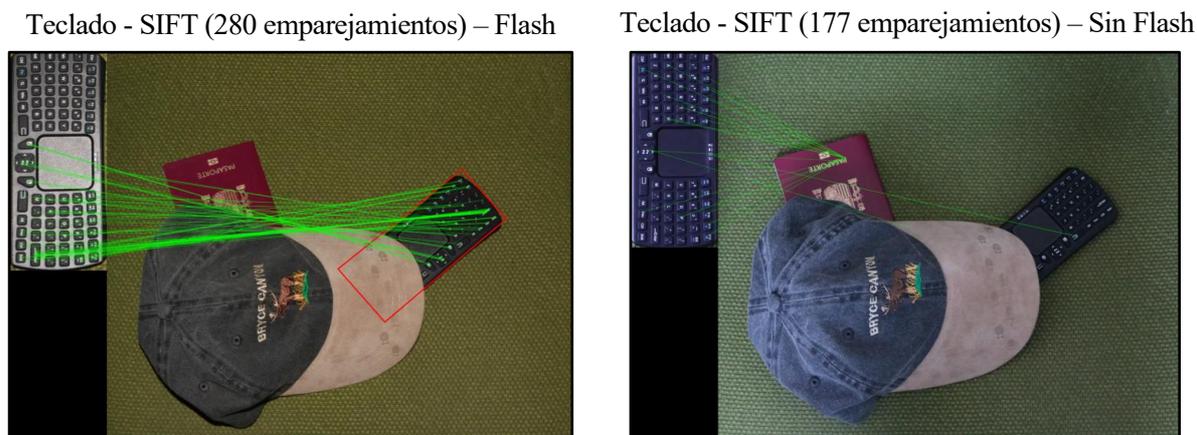


Figura 6.11. Identificación de objetos ocultos parcialmente con y sin fuente de luz.  
[Fuente: Elaboración propia]

Se puede concluir que el uso de una fuente de luz adecuada es conveniente en cualquiera de las situaciones ya que consigue destacar más información de la imagen y los métodos implementados pueden detectar y describir más puntos característicos. La intensidad de dicha fuente de luz debe ajustarse adecuadamente para evitar deslumbramientos y pérdida de detalles.

En cuanto a los tiempos de ejecución, ambos métodos tardan entre 3 y 6 segundos en un portátil con un procesador i7-5500U, 8 GB de RAM y sin tarjeta gráfica dedicada (GPU). Además, las imágenes usadas son de una resolución muy alta (2800 x 2400 píxeles aproximadamente). Estos datos son orientativos ya que para realizar un análisis de tiempo de ejecución se deben analizar muchas más imágenes en diferentes condiciones.

No obstante, en la solución real no se necesitan tiempos de ejecución rápidos puesto que el número de movimientos de piezas en las estanterías a lo largo del día es pequeño. Esta situación estacionaria es mucho mayor que el tiempo computacional de estos métodos.

Este proyecto pretende comprobar la viabilidad de la implantación de la solución de visión artificial. En el caso de implantar dicho programa en una situación industrial real, se deben realizar muchas más pruebas que las mostradas en este documento. Estas deben contemplar los cambios de iluminación, la resolución de las cámaras usadas, los tiempos requeridos, dimensiones de la estantería, colores del interior de las cajas, etc. Además, se deben acotar los tipos de piezas y herramientas a detectar ya que, por ejemplo, en el caso de piezas pequeñas la cantidad de puntos característicos encontrados puede no ser suficiente para su identificación, y a la vez se pueden confundir sus descriptores con los de otras piezas. La realización de estas pruebas no ha sido posible hasta el momento, por lo que el único dato real del que se dispone es una imagen que contiene un material. Para esta imagen, se obtienen los resultados de la Figura 6.12.

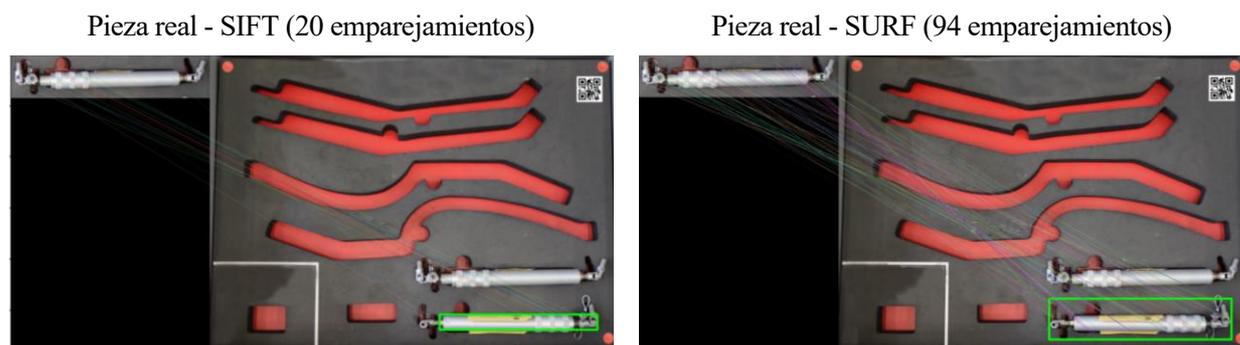


Figura 6.12. Identificación de objeto real con SIFT y SURF. [Fuente: Elaboración propia]

El alcance de este proyecto reside en mostrar la viabilidad del uso de esta tecnología para dar solución al problema de localización de materiales. Por tanto, estos resultados no son suficientes para asegurar un funcionamiento real correcto de los métodos de localización propuestos como solución. Como se ha comentado, antes de implantar estos métodos se deberían realizar las diferentes pruebas en dicho entorno real, en condiciones y situaciones variadas.

Finalmente, se han de tener en cuenta los posibles errores que pueden surgir con su implementación, ya que estos métodos pueden indicar erróneamente la existencia de un objeto sin que este se encuentre, y viceversa. Por este motivo se debe adaptar el programa para posibilitar la corrección manual.

# 7 CONCLUSIONES Y LÍNEAS FUTURAS

---

Partiendo de la necesidad de conocer la localización de los materiales y la inexistencia de datos digitales, se ha conseguido desarrollar un sistema mediante visión artificial, que permite la automatización de esta tarea. Este proyecto es la base principal para las mejoras y ampliaciones futuras por parte de la empresa AIRBUS, y para ello se han usado únicamente funciones de alto nivel que ofrece la librería OpenCV con el lenguaje de programación Python. Así pues, mediante el uso de procesamiento de imágenes y extracción de características locales se ha conseguido comprobar la viabilidad de esta solución propuesta mediante los códigos y las pruebas del capítulo 5 y 6.

Se puede afirmar que el uso de la visión artificial para la localización de materiales proporciona buenas soluciones a la mayoría de los problemas de este tipo, incluso pudiendo mejorar con el rediseño de la caja y estanterías propuestos. Además, como se ha comentado en el capítulo anterior, el uso de una correcta iluminación haciendo uso de una fuente de luz artificial mejora considerablemente los resultados obtenidos.

Adicionalmente cabe destacar que el coste del rediseño de estos elementos y de la implantación del sistema sería amortizada en poco tiempo ya que se ahorrarían grandes tiempos de proceso, y por tanto de entrega.

No obstante, no se puede confiar todo en esta tecnología, siendo muy recomendable hacer una unión de RFID y visión artificial, y tratar los datos en conjunto. Además, se debe tener en cuenta que se deben diseñar por completo todos los aspectos del programa, como la correcta captura de las imágenes de la base de datos, la adecuación a todas las cámaras de las estanterías, etc., para su correcta implantación en la industria.

Como este proyecto se centra en la empresa AIRBUS y puede ser continuado posteriormente, a continuación, se proponen varias líneas futuras que se podrían seguir con el objetivo de mejorar los resultados y aplicar la solución propuesta a otros ámbitos:

- Probar diferentes métodos de extracción de características locales como ORB y AKAZE para comparar sus resultados con SIFT y SURF.
- Diseñar nuevos métodos basados en SIFT para encontrar las características más adecuadas para las piezas y herramientas reales (bordes, texturas, etc.).
- Uso de imágenes con menor resolución. Esto debería disminuir el número de puntos característicos encontrados, pero mejoraría en cuando a tiempo computacional. Sería interesante su estudio cuando se requiera una minimización del tiempo de detección y localización de las piezas.
- Implantación de otro sistema de visión en la llegada de los paquetes a las instalaciones, como se comentó en el capítulo 3. De esta forma las imágenes de las piezas pueden ser tomadas y guardadas automáticamente para usarse de referencia en el programa de este documento. Además, consigue la automatización del proceso, la información digital al instante y la eliminación de errores humanos, proporcionando a cada pieza una alta trazabilidad.
- Diseño completo de las estanterías, siendo conveniente incorporar una red de distribución de datos a través de la red IoT de la empresa. Esto permite disponer de toda la información en todos los dispositivos e incorporar la información en el resto de las aplicaciones corporativas.
- Ampliación de funciones del programa actual, de manera que pueda ser tratada la información por la red IoT. Es conveniente el uso de fuentes de luz, sensores de proximidad, entre otros, y programar estas conexiones de manera adecuada.



# REFERENCIAS

---

- Adel, E. (2019). Introduction to Image Processing. Retrieved from [https://sbme-tutorials.github.io/2019/intro-to-BME/presentations/12\\_week12/index.html#21](https://sbme-tutorials.github.io/2019/intro-to-BME/presentations/12_week12/index.html#21)
- AllGo Vision. (n.d.). *Face Recognition Software with High Accuracy - AllGoVision*. Retrieved from <https://www.allgovision.com/face-recognition.php>
- ATOX Sistemas de almacenaje. (2017, Julio 05). *Inventario fantasma y cómo combatirlo*. Retrieved from ATOX Sistemas de almacenaje: <http://www.atoxgrupo.com/website/noticias/inventario-fantasma>
- BCM Families Foundation. (n.d.). *El ojo – Blue Cone Monochromacy*. Retrieved from <http://www.blueconemonochromacy.org/es/how-the-eye-functions/>
- Braun, J., & Flores, P. (2011, Agosto 9). Algoritmo SIFT: fundamento teorico.
- Cabrera Ibarra, H., Codratian Rosu, H., Torres González, L. A., & Treviño Gutiérrez, J. P. (2007). La codificación de los quipus incas. (A. M. Ciencias, Ed.) *Ciencia*, 58(4), 26-33.
- EASYCONTROL, S.L. (n.d.). *Easycontrol - Advanced Vision Machine Systems For Control Parts*. Retrieved from <https://easycontrol.net/>
- Europa Press. (2019, Marzo 14). *Aena prepara para este mes una prueba piloto de reconocimiento facial en el aeropuerto de Menorca*. Retrieved from Menorca - Es diari: <https://www.menorca.info/menorca/local/2019/03/11/651821/aena-prepara-para-este-mes-prueba-piloto-reconocimiento-facial-aeropuerto-menorca.html>
- FQ Ingeniería Electrónica. (n.d.). *FQ Ingeniería Electrónica*. Retrieved from <https://www.fqingenieria.com/>
- Khattab, A., Jeddi, Z., Amini, E., & Bayoumi, M. (2017). *RFID Security. A Lightweight Paradigm*. Springer.
- Luke, J. (2018, Agosto 5). *SIFT Object Matching in Python*. Retrieved from Stack Overflow: <https://stackoverflow.com/questions/51693427/sift-object-matching-in-python>
- Maravall Gómez Allende, D. (1993). *Reconocimiento de formas y visión artificial*. Madrid: Ra-Ma.
- Ministerio de Educación. (2012, Febrero). Aplicación práctica de la visión artificial.
- National Geographic en español. (2018, Agosto 09). *National Geographic en Español*. Retrieved from Los quipus, la escritura secreta de los antiguos incas: <https://www.ngenespanol.com/el-mundo/que-es-un-quipu-escritura-inca-epoca-precolombina/>
- Neosentec. (2017, Diciembre 07). *7 Ventajas de la realidad aumentada en empresas*. Retrieved from Neosentec: <https://www.neosentec.com/7-ventajas-realidad-aumentada-empresas/>
- OpenCV. (n.d.). *Open CV - Open Source Computer Vison*. Retrieved from <https://docs.opencv.org/master/index.html>
- Rehman, A. (2020, Abril 09). *Real Time Contours Detection findcontours OpenCV Python*. Retrieved from FYP Solutions: <http://www.fypsolutions.com/opencv-python/findcontours-opencv-python-drawcontours-opencv-python/>

- Sossa Azuela, J. H. (2013). *Visión artificial : rasgos descriptores para el reconocimiento de objetos*. Madrid: Ra-Ma.
- Stack Overflow. (2019, Noviembre 29). *How to read electricity meter using Computer Vision + Tesseract?* Retrieved from Stack Overflow: <https://stackoverflow.com/questions/59102715/how-to-read-electricity-meter-using-computer-vision-tesseract>
- Taibani, I. (2019, Septiembre 18). *Basic Image Processing Using MATLAB*. Retrieved from Electronics For You: <https://www.electronicsforu.com/electronics-projects/software-projects-ideas/image-processing-using-matlab>
- Torres Gutiérrez, J., Rivera Retamar, R., & Valero Gómez, J. J. (2018). Una imagen vale más que mil palabras.
- Vargas Villanueva, M. (n.d.). Temario asignatura "Sistemas de Percepción", Escuela Técnica Superior de Ingeniería. Sevilla, España.
- Witte printed technology. (n.d.). *RFID labels from experienced specialists | Witte Group*. Retrieved from <https://www.witte-group.de/en/produkte/rfid-label-solutions-plusident/>