



## **GRADO EN INGENIERÍA DE TECNOLOGÍAS INDUSTRIALES**

Escuela Superior de Ciencias Experimentales y Tecnología

Curso académico 2024-2025

### **Trabajo Fin de Grado**

Sistema de reconocimiento por visión de maduración de frutos  
para su recolección con un brazo robótico

**Autor:** David Campoamor Medrano

**Tutor:** Julio Vega Pérez



Este trabajo se distribuye bajo los términos de la licencia internacional CC BY-NC-SA International License (Creative Commons AttributionNonCommercial-ShareAlike 4.0). Usted es libre de *(a) compartir*: copiar y redistribuir el material en cualquier medio o formato; y *(b) adaptar*: remezclar, transformar y crear a partir del material. El licenciador no puede revocar estas libertades mientras cumpla con los términos de la licencia:

- *Atribución.* Usted debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciatante.
- *No comercial.* Usted no puede hacer uso del material con propósitos comerciales.
- *Compartir igual.* Si remezcla, transforma o crea a partir del material, debe distribuir su contribución bajo la misma licencia del original.

*Documento de David Campoamor Medrano.*

# Agradecimientos

---

Nunca es tarea fácil agradecer a tantas personas el apoyo, la ayuda y los consejos que han contribuido en mi beneficio, tanto personal como académico, durante todos estos años.

En primer lugar, me gustaría dar las gracias tanto a la Universidad Rey Juan Carlos como a todos los profesores de los que he tenido el privilegio de ser alumno, por haber sido capaces de transmitir la dedicación, pasión, disciplina y el esfuerzo tan imprescindible como necesarios para la praxis de una profesión como lo es la de ingeniero, y más concretamente en mi caso, la de ingeniero industrial.

Quisiera expresar mi gratitud a mi tutor, Julio Vega, por guiarme, acompañarme y ayudarme durante estos meses de trabajo, para mí fue todo un honor saber que finalmente había aceptado dirigir este trabajo final de grado, y de este modo cerrar un bonito círculo que empezó con él como profesor mío de informática en el colegio, donde nos enseñó, entre otras muchas cosas, que más allá de los editores de texto convencionales, existen otros sistemas para la preparación de documentos, por esto, este trabajo también es en parte suyo, ya que tanto estas líneas como el resto del documento están basados en sus enseñanzas.

Asimismo, me gustaría agradecer a Robotplus, por cumplimentar mi formación académica y darme mi primera oportunidad laboral en el ámbito industrial, y más concretamente a mis compañeros del departamento de servicio técnico y a los del departamento de I+D+i, ya que gracias a ellos hoy por hoy he podido entender y experimentar más en profundidad muchos de los principios teóricos y de los problemas que únicamente conocía sobre el papel, pudiendo desarrollarme de una manera más completa como profesional.

Agradecer también a mis amigos y compañeros de clase, los *Hijos de la Ingeniería* y David, por no haber dejado que me rindiera incluso en los peores momentos y con todo en contra, y por haber sido un gran apoyo tanto dentro como fuera de la universidad.

A mis amigos del equipo de baloncesto en Alcorcón, en especial a Rober y a Adri, por haber confiado siempre en que este momento llegaría, antes o después, y haber formado parte de este proceso del que desde antes de empezar la universidad ya formaban parte, al igual que mis amigos de Móstoles del colegio, el *Cártel de La Manga*. Y sobre todo, gracias a Sandra, por ser para mí el claro ejemplo de que la dedicación y el trabajo duro merecen la pena, pero más allá de todo esto, por estar a mi lado día a día y ser mi compañera de vida, sin ella no habría podido soñar con finalmente llegar hasta aquí.

No querría concluir los agradecimientos sin hacer partícipe a toda mi familia, y en especial a mis padres y mi hermano, la paciencia que han tenido todo este tiempo conmigo, sobre todo en época de entregas y de exámenes, pero sobre todo y más importante, la confianza depositada en mí, que mediante palabras y gestos de apoyo incondicional han demostrado. Ha sido gracias a este amor y apoyo que solo la familia sabe darte cuando más lo necesitas, por lo que sido más fácil poder alcanzar esta meta. Gracias a mis tíos y a mis primos mayores, por hacer que me interesase en el mundo de las ciencias, y más concretamente en la ingeniería y la construcción, faceta en la que ya desde pequeño había fijado mi atención jugando con aquellos bloques fabricados en plástico ABS y de colorines, ya que sin duda, fue gracias a ellos por lo que terminé de decidir embarcarme, ya desde el colegio, en las materias que guardaban mayor similitud con estos aspectos antes que en otras, puesto que veía en ellos una referencia a seguir. Pero sobre todo, gracias a mis abuelos, que como suele decirse, deberían ser eternos. Si antes hablaba de referencias, sin duda ellos han sido el máximo exponente en esto, puesto que sin sus enseñanzas y consejos, y no solo en aspectos académicos, no podría haber llegado hasta aquí. Todos ellos siempre formarán parte de mi y estarán presentes en cada una de las tomas de decisiones importantes que tenga que llevar a cabo, en las desilusiones y en los malos ratos, pero también en la consecución de mis éxitos y logros, como es el caso, aunque algunos de ellos ya no se encuentren entre nosotros o no puedan recordarlo. Espero haber podido aprender y retener algo de la sabiduría que me habéis mostrado y trasmitido.

*A todas aquellas personas que, con trabajo y esfuerzo, terminan consiguiendo todo aquello que se proponen.*

Madrid, xx de xxxxxxx de 20xx

*David Campoamor Medrano*

# Resumen

---

Escribe aquí el resumen del trabajo. Un primer párrafo para dar contexto sobre la temática que rodea al trabajo.

Un segundo párrafo concretando el contexto del problema abordado.

En el tercer párrafo, comenta cómo has resuelto la problemática descrita en el anterior párrafo.

Por último, en este cuarto párrafo, describe cómo han ido los experimentos.

# **Abstract**

---

Escribe aquí el resumen del trabajo. Un primer párrafo para dar contexto sobre la temática que rodea al trabajo.

Un segundo párrafo concretando el contexto del problema abordado.

En el tercer párrafo, comenta cómo has resuelto la problemática descrita en el anterior párrafo.

Por último, en este cuarto párrafo, describe cómo han ido los experimentos.

# Acrónimos

---

**ABB** *Asea Brown Boveri*

**AER** *Asociación Española de Robótica*

**AERO** *Autonomous Exploration Rover*

**AGV** *Automated Guided Vehicle*

**AI** *Artificial Intelligence*

**AMR** *Autonomous Mobile Robot*

**ANN** *Artificial Neural Network*

**API** *Application Programming Interface*

**CMI** *Cirugía Mínimamente Invasiva*

**CPU** *Central Processing Unit*

**dFoV** *diagonal Field of View*

**DL** *Deep Learning*

**DLR** *Centro Aeroespacial Alemán (Deutsches Zentrum für Luft - und Raumfahrt e. V.)*

**DNN** *Deep Neural Network*

**DOF** *Degree of Freedom*

**EKF** *Extended Kalman Filter*

**EPFL** *Escuela Politécnica Federal de Lausana*

**FDA** *Administración de Alimentos y Medicamentos de EE.UU. (Food and Drug Administration)*

**FOA** *Focus of Attention*

**FPS** *Fotogramas por Segundo*

**GA** *Genetic Algorithm*

**GPIO** *General Purpose Input/Output*

**GPS** *Global Positioning System*

**HCI** *Human-Computer Interaction*

**HRI** *Human-Robot Interaction*

**IA** *Inteligencia Artificial*

**IBM** *International Business Machines*

**IFR** *International Federation of Robots*

**IP** *Ingress Protection*

**ISO** *Internacional Organization for Standardization*

**LTS** *Long Term Support*

**LWR** *Lightweight Robot*

**ML** *Machine Learning*

**NN** *Neural Network*

**OSRF** *Open Source Robotics Foundation*

**PE** *Process Element*

**PUMA** *Programmable Universal Machine for Assembly*

**RNA** *Redes Neuronales Artificiales*

**ROS** *Robot Operating System*

**ROS-I** *Robot Operating System-Industrial*

**SAIL** *Stanford Artificial Intelligence Laboratory*

**SCARA** *Selective Compliance Assembly Robot Arm*

**SCB** *Safety Control Board*

**SML** *Shallow Machine Learning*

**SRI** *Stanford Research Institute*

**TC** *Technical Committee*

**UR** *Universal Robots*

**UWB** *Ultra-Wideband*

**VA** *Visión Artificial*

**YOLO** *You Only Look Once*

# Índice general

---

<b>1. Introducción</b>	<b>1</b>
1.1. Robots y robótica . . . . .	2
1.1.1. Robots industriales . . . . .	3
1.1.2. Robots de servicio . . . . .	13
1.1.3. Robot Médico . . . . .	19
1.2. Inteligencia Artificial . . . . .	21
1.3. Visión Artificial . . . . .	23
1.4. Machine Learning . . . . .	25
1.5. Deep Learning . . . . .	27
<b>2. Estado del arte</b>	<b>30</b>
<b>3. Objetivos</b>	<b>40</b>
3.1. Descripción del problema . . . . .	40
3.2. Requisitos . . . . .	41
3.3. Competencias . . . . .	42
3.4. Metodología . . . . .	44
3.5. Plan de trabajo . . . . .	46
<b>4. Plataforma de desarrollo</b>	<b>48</b>
4.1. Hardware . . . . .	48
4.1.1. Cámara Logitech C270 HD . . . . .	48
4.1.2. Soporte de brazo articulado . . . . .	49
4.1.3. Ordenador principal . . . . .	49
4.1.4. Robot de Universal Robots de la gama e-series . . . . .	51
4.2. Software . . . . .	52
4.2.1. Ubuntu . . . . .	52
4.2.2. Polyscope . . . . .	54
4.2.3. Python . . . . .	54

4.2.4. PyTorch . . . . .	54
4.2.5. NumPy . . . . .	54
4.2.6. OpenCV . . . . .	55
4.2.7. XML-RPC . . . . .	56
4.2.8. Anaconda . . . . .	56
4.2.9. YOLOv3 . . . . .	57
<b>5. Diseño . . . . .</b>	<b>59</b>
5.1. Snippets . . . . .	59
5.2. Verbatim . . . . .	59
5.3. Ecuaciones . . . . .	60
5.4. Tablas o cuadros . . . . .	60
<b>6. Conclusiones . . . . .</b>	<b>62</b>
6.1. Conclusiones . . . . .	62
6.2. Corrector ortográfico . . . . .	63
<b>Bibliografía . . . . .</b>	<b>64</b>

# Índice de figuras

---

1.1.	Primer robot industrial . . . . .	4
1.2.	Standford Arm . . . . .	5
1.3.	Diagrama mecatrónico de construcción de máquinas inteligentes . . . . .	5
1.4.	Robot Cincinnati Milacron T3 . . . . .	6
1.5.	Uno de los primeros prototipos de robot SCARA . . . . .	7
1.6.	Robot ABB IRB 360 Flexpicker . . . . .	7
1.7.	Robot Motoman DA-20 . . . . .	9
1.8.	LWR3 . . . . .	9
1.9.	Robots utilizados para el desarrollo de ROS . . . . .	10
1.10.	Logo de ROS . . . . .	10
1.11.	UR5 con su controladora . . . . .	11
1.12.	Universal Robots e-Series . . . . .	12
1.13.	Robot Shakey . . . . .	13
1.14.	Robots de servicio . . . . .	15
1.15.	Robot aspirador Roomba de iRobot . . . . .	15
1.16.	Robots de inspección y mantenimiento . . . . .	16
1.17.	Robots de educación . . . . .	16
1.18.	Agricultura vertical . . . . .	17
1.19.	Robots de logística . . . . .	18
1.20.	Robots de entretenimiento . . . . .	18
1.21.	Robot Da Vinci . . . . .	19
1.22.	Robots médicos . . . . .	20
1.23.	Modelos de inteligencia . . . . .	22
1.24.	Aspectos clave y semántica del Aprendizaje automático . . . . .	25
1.25.	Diagrama de Venn de la relación entre distintas áreas de la IA . . . . .	26
1.26.	Modelo biológico de una neurona genérica, comúnmente una neurona motora típica (izquierda) y el respectivo modelo matemático (derecha)	
1.27.	Arquitectura de una red neuronal . . . . .	28

2.1. Representación de varios tipos de robots agrícolas . . . . .	30
2.2. Ilustración global del rendimiento general de los robots revisados . . . . .	31
2.3. Agrobot . . . . .	33
2.4. Diseño conceptual del robot de recogida con sus componentes . . . . .	34
2.5. Robot agrícola Dogtooth . . . . .	35
2.6. Montaje del hardware en una explotación de fresas: el robot consta principalmente de una cámara RGB-D, un brazo 5-DOF, una pinza y una plataforma móvil . . . . .	36
2.7. Detección de tomates indicada por los círculos rojos . . . . .	38
2.8. Robot recolector de tomates . . . . .	39
3.1. Ciclo de la metodología DMADV . . . . .	45
4.1. Cámara Logitech C270 HD <sup>16</sup> . . . . .	48
4.2. Soporte de brazo articulado <sup>17</sup> . . . . .	49
4.3. Lenovo Legion 5 15IMH05 <sup>18</sup> . . . . .	50
4.4. Gama e-series de Universal Robots <sup>19</sup> . . . . .	52
4.5. Lanzamiento de las versiones de Ubuntu . . . . .	53
4.6. Logo de Python . . . . .	54
4.7. Logo de YOLO . . . . .	57

# Listado de códigos

---

5.1. Función para buscar elementos 3D en la imagen . . . . .	59
5.2. Cómo usar un Slider . . . . .	60

# Listado de ecuaciones

---

5.1. Ejemplo de ecuación con fracciones . . . . .	60
5.2. Ejemplo de ecuación con array y letras y símbolos especiales . . . . .	60

# Índice de cuadros

---

1.1. Procesos de la visión artificial . . . . .	23
4.1. Especificaciones técnicas del ordenador usado . . . . .	50
5.1. Parámetros intrínsecos de la cámara . . . . .	61

---

# **Capítulo 1**

## **Introducción**

---

Desde sus inicios, la robótica ha proporcionado un sinfín de posibilidades y alternativas ante problemas que anteriormente carecían de las soluciones adecuadas, pero, ¿qué es realmente la robótica?

Se podría definir robótica como el proceso mediante el cual una máquina intercambia energía e información con su entorno, con el propósito de alcanzar una serie de objetivos específicos. Este campo tecnológico en expansión es el resultado de décadas de colaboración continua entre biólogos, informáticos e ingenieros [Koditschek, 2021]. Dada esta multidisciplina, la robótica abarca una amplia gama de aplicaciones, desde la industria hasta la medicina, pasando por la exploración espacial, la domótica o la conducción autónoma, entre otras. Es un campo en constante evolución, impulsado por la búsqueda de soluciones innovadoras para mejorar la calidad de vida y permitir superar desafíos de manera más eficiente y segura.

La industria agrícola no es una excepción, ya que ha contemplado históricamente tareas que requieren una dedicación laboral considerable. No obstante, gracias a la robótica y a los sistemas de visión artificial, surge la oportunidad de transformar una serie de procesos, como puede ser la recolección de cultivos a través de la detección automatizada.

En las siguientes secciones describiremos brevemente algunas de las aplicaciones más importantes de la robótica en la sociedad actual, así como los distintos conceptos en los cuales se basa la investigación y el desarrollo llevado a cabo.

## 1.1. Robots y robótica

Según la Federación Internacional de Robots (IFR) se define robot según el vocabulario establecido por la International Organization for Standardization (ISO), y esto es como *mecanismo accionado programado con cierto grado de autonomía para realizar tareas de locomoción, manipulación o posicionamiento* [ISO/TC299, 2021].

El término robot fue utilizado por primera vez por Karel Capek (en su obra de teatro Rossum's Universal Robots publicada en 1920. Esta palabra viene del vocablo checo “*robota*” que significa trabajo, en el sentido de la obligatoriedad, entendido como servidumbre, trabajo forzado o esclavitud [Sánchez Martín et al., 2007a]. Aunque esta definición es un punto de partida, es cierto que es posible diferir en aspectos como si un robot debe controlarse automáticamente o podría ser autónomo o si un robot debe ser reprogramable. A un nivel más amplio, cualquier máquina que pueda utilizarse para llevar a cabo acciones o tareas complejas de forma automática puede considerarse un robot [Raj and Seamans, 2019].

Históricamente, las civilizaciones antiguas, como la egipcia y la griega, dieron los primeros pasos en lo que se puede denominar robótica clásica, construyendo autómatas y mecanismos diseñados para imitar acciones humanas, con características mecánicas rudimentarias. Con el paso del tiempo, la ciencia y la ingeniería avanzaron, y los conceptos de la robótica comenzaron a tomar forma más definida hasta que, en el siglo XX, con el desarrollo de la ingeniería en sus diferentes ramas (mecánica, electrónica, informática, telecomunicaciones), Isaac Asimov (1920-1992) utilizó por primera vez el término robótica y postuló las tres leyes de la robótica en su libro *I Robot* publicado en 1950, coincidiendo con el apogeo de la robótica moderna. Asimov consideró necesario añadir una cuarta ley, antepuesta a las demás, la número cero, que afirma que un robot no debe actuar simplemente para satisfacer intereses individuales, sino que sus acciones deben preservar el beneficio común de toda la humanidad [Sánchez Martín et al., 2007b].

Es también en 1950 cuando Alan Mathison Turing publica *Computing Machinery and Intelligence* y propone una prueba (test o máquina de Touring), en forma de entidad matemática abstracta, que demuestra la existencia de problemas computacionales irresolubles que ninguna máquina es capaz de solventar. Se puede afirmar que un programa de ordenador no llegará nunca a ser tan inteligente como un ser humano y que un robot no podrá suplir al ser humano de forma completa [Sánchez Martín et al., 2007b]

preocupación sobre el potencial de sustitución de la mano de obra, que históricamente ha atenuado el entusiasmo en torno a las nuevas tecnologías [Mokyr et al., 2015].

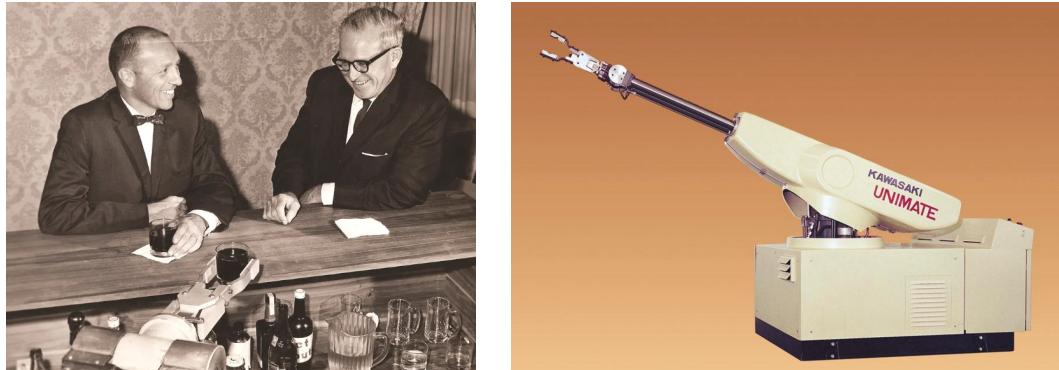
Partiendo de todos estos avances y del interés por automatizar las tareas de producción, la robótica va adquiriendo un gran desarrollo [Sánchez Martín et al., 2007b]. Es debido a este desarrollo que, atendiendo al propósito y al contexto en el que se utilicen estos robots, se fueron creando varios grupos en función de los que clasificarlos. Estos tres grandes grupos fueron, en función de una serie de criterios generales: robots industriales, robots de servicio y robots médicos.

### 1.1.1. Robots industriales

Se define robot industrial como un manipulador polivalente, reprogramable y controlado automáticamente, programable en tres o más ejes, que puede ser fijo o móvil para su uso en aplicaciones de automatización industrial [ISO/TC299, 2021].

El inicio de la robótica industrial puede datarse en la década de 1950, aunque algunos tipos de automatización en el entorno industrial empezaron a aparecer desde los tiempos de la Revolución Industrial. La evolución de los robots industriales puede subdividirse en cuatro categorías: las tres primeras abarcan el período comprendido entre los años cincuenta y finales de los noventa, mientras que la cuarta generación abarca desde 2000 hasta nuestros días [Gasparetto and Scalera, 2019].

La primera generación, o primeros manipuladores (1950-1967), eran básicamente máquinas programables que no tenían comunicación con el entorno externo y con algoritmos de control sencillos (punto a punto). En cuanto al hardware, contaban con equipos de baja tecnología, sin servo-controladores. Sin embargo, en 1954, George Devol y Joseph Engelberger formaron la empresa Unimation, empresa que desarrollaría Unimate (ver en Figura 1.1), considerado el primer robot industrial de la historia, fabricado en 1961 [Zamalloa et al., 2017].



(a) Joseph Engelberger y George Devol

(b) Robot Unimate

Figura 1.1: Primer robot industrial

Empresas como Ford y General Motors empezaron a plantearse la automatización de sus plantas productivas, por lo que, en 1962, la empresa AMF Corporation fabricó un nuevo robot llamado Versatan, un robot cilíndrico que Ford encargó para sus fábricas. Este robot, fue también el primero que se instaló en un centro productivo en Japón [Gasparetto and Scalera, 2019].

La segunda generación, o robots sensorizados (1968-1977), eran máquinas programables básicas con posibilidades limitadas de comportamiento autoadaptativo y capacidades elementales para reconocer el entorno externo, poseían sistemas sensoriales avanzados y eran robots de gran volumen que se utilizaban principalmente en automoción [Zamalloa et al., 2017].

En 1968, en el Stanford Artificial Intelligence Laboratory (SAIL) se confecciona el WAVE, el primer lenguaje de programación para robots. En 1969, Unimation concedió a Kawasaki Heavy Industries Ltd. la licencia para producir robots para el mercado japonés y asiático, conduciendo al desarrollo del Kawasaki-Unimate 2000, el primer robot industrial construido en Japón. Es también en este año cuando Víctor Scheinman, un estudiante de ingeniería mecánica de la Universidad de Standford, diseñó y construyó el primer prototipo de brazo robótico (Figura 1.2), cuya cinemática inversa podía resolverse de manera analíticamente cerrada, permitiendo una rápida ejecución de la trayectoria [Gasparetto and Scalera, 2019].

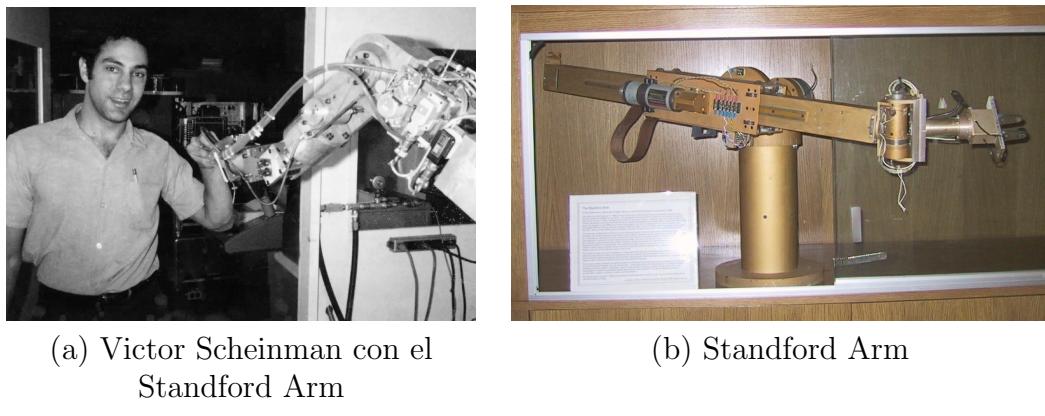


Figura 1.2: Stanford Arm

En 1969, el ingeniero de la compañía Yaskawa, T Mori, acuña el término mecatrónica, que integra el conjunto de mecanismos de control automático imprescindibles para el desarrollo de cualquier máquina inteligente [Sánchez Martín et al., 2007b] tal y como muestra el diagrama de la Figura 1.3.

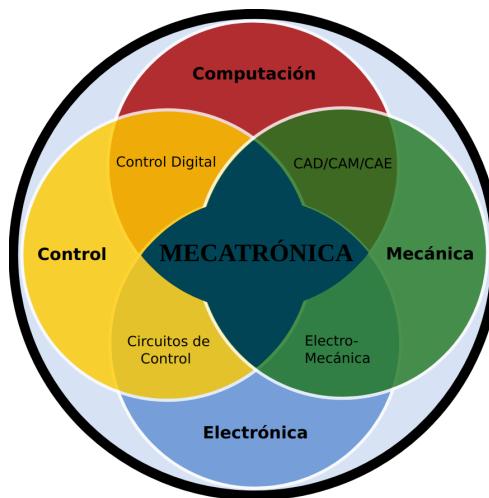


Figura 1.3: Diagrama mecatrónico de construcción de máquinas inteligentes

En 1973, KUKA<sup>1</sup> construyó el primer robot industrial con 6 ejes electromecánicos llamado Famulus. Un año más tarde, Cincinnati Milacron introdujo en el mercado el robot T3 (Figura 1.4). Cincinnati Milacron (adquirida por ABB<sup>2</sup> en 1990). El robot T3 fue el primer robot comercial controlado por un microordenador [Zamalloa et al., 2017].

<sup>1</sup><https://www.kuka.com/es-es>

<sup>2</sup><https://new.abb.com/products/robotics>



Figura 1.4: Robot Cincinnati Milacron T3

La tercera generación, o robots industriales (1978-1999), disponían de controladores específicos (ordenadores), siendo un punto clave en la caracterización de esta generación, además del surgimiento de nuevos lenguajes de programación para el control de los robots, la posibilidad de reprogramarlos y la inclusión parcial de la visión artificial [Zamalloa et al., 2017]. A finales de los años setenta y principios de los ochenta, otros avances científicos y técnicos contribuyeron a la difusión de los robots [Gasparetto and Scalera, 2019], que junto a que las empresas de todo el mundo invirtieron miles de millones de dólares en del mundo para automatizar tareas básicas en sus cadenas de montaje, supusieron que los robots poblaran muchos sectores industriales para automatizar una amplia variedad de actividades [Zamalloa et al., 2017].

En 1978, Unimation diseñó y fabricó el robot PUMA. El PUMA (*Programmable Universal Machine for Assembly*) fue considerado durante muchas décadas el arquetípico de los robots antropomórficos [Gasparetto and Scalera, 2019]. Ese mismo año, el científico japonés Hiroshi Makino, de la Universidad de Yamanashi, propuso una nueva estructura cinemática. El robot con esta estructura se denominó SCARA (*Selective Compliance Assembly Robot Arm*) (ver Figura 1.5), ya que su conformidad en la dirección horizontal resultó menor que la conformidad en la dirección vertical. Por esta razón, así como por la ligereza de la cadena cinemática (que permitía un controlador más sencillo y rápido), este robot era adecuado para ser empleado en tareas como el ensamblaje de objetos pequeños [Makino and Furuya, 1980].



Figura 1.5: Uno de los primeros prototipos de robot SCARA

Más tarde, en 1981, en la Universidad Carnegie-Mellon se desarrolló un robot de impulsión directa que utiliza motores eléctricos en las articulaciones, evitando la distorsión de las transmisiones mecánicas convencionales. En 1982, IBM introduce el robot de montaje industrial RS-1 que utiliza un brazo constituido por 3 dispositivos de deslizamiento [Sánchez Martín et al., 2007b]. De la idea de emplear cadenas cinemáticas paralelas en lugar de las clásicas cadenas cinemáticas en serie, junto con la de crear un robot ligero capaz de moverse a gran velocidad, surgió el arquetipo del robot Delta (que apareció en 1992), concebido por el científico suizo Reymond Clavel en la Escuela Politécnica Federal de Lausana (EPFL) [Clavel, 1991]. En comparación con los robots en serie, los robots paralelos tienen un espacio de trabajo más pequeño, pero pueden funcionar a una velocidad mucho mayor, siendo la arquitectura cinemática ideal para los robots dedicados a operaciones de pick-and-place de alta velocidad. Basado en este tipo de estructura, unos años después, en 1998, ABB desarrolló el Flex-Picker (Figura 1.6, el robot de picking más rápido del mundo [Gasparetto and Scalera, 2019]).



Figura 1.6: Robot ABB IRB 360 Flexpicker

A partir de los años 2000, aparece la cuarta generación o robots inteligentes (2000-Actualidad), que se caracteriza por la inclusión de capacidades informáticas avanzadas, ya que los ordenadores no sólo trabajan con datos, si no también pueden realizar razonamientos lógicos y aprender, puesto que la Inteligencia Artificial comienza a ser incluida parcial y experimentalmente en estos robots. Los sensores son más sofisticados, y envían información al controlador y la analizan mediante estrategias de control complejas para que el robot pueda basar sus acciones en información sólida y fiable. Es en esta generación cuando se introducen los robots colaborativos [Zamalloa et al., 2017].

Los requisitos de velocidad y peso de un robot han dado lugar a novedosos diseños cinemáticos y de transmisión. Desde el principio, la reducción de la masa y la inercia de las estructuras robóticas ha sido un objetivo primordial en el desarrollo de la robótica. El brazo humano, con una relación peso-carga de 1:1, se consideraba la referencia definitiva [Siciliano and Khatib, 2016]. Con este objetivo se desarrollaron, gracias a la colaboración entre la empresa alemana KUKA y el Instituto de Robótica y Mechatrónica del Centro Aeroespacial Alemán (DLR), tres generaciones de robots ligeros, permitiendo a investigadores e ingenieros desarrollar nuevas aplicaciones de robótica industrial y de servicios con un rendimiento sin precedentes. En el año 2004, con motivo de Automática, la mayor exposición de robots del mundo, se presentó por primera vez la combinación entre el robot ligero del DLR y la controladora KUKA, denominado RoboAssistant, donde se permitió a los visitantes mover y programar manualmente el robot, haciendo la visión de un robot que asiste a un trabajador durante los procesos de producción evidente para los visitantes [Bischoff et al., 2010].

Es en estos procesos de producción, donde la manipulación a dos manos puede ser crítica para tareas de ensamblaje complejas, manipulación simultánea y procesamiento de piezas de trabajo o para la manipulación de objetos de gran tamaño, por lo que en 2005, MOTOMAN presenta el primer robot comercial para la manipulación sincronizada a dos manos [Siciliano and Khatib, 2016] (ver Figura 1.7).



Figura 1.7: Robot Motoman DA-20

Sin embargo, fue en el año 2006 cuando, tras haber estado trabajando en la búsqueda de vías para el desarrollo en serie del robot LWR3 del DLR, y tras una intensa cooperación entre KUKA y el DLR para transmitir a los desarrolladores de KUKA los conocimientos necesarios para el desarrollo de brazos ligeros, componentes y electrónica integrada, cuando se toma la decisión de producir una primera pequeña serie del robot ligero del KUKA LWR3 [Bischoff et al., 2010], tal y como muestra la Figura 1.8.

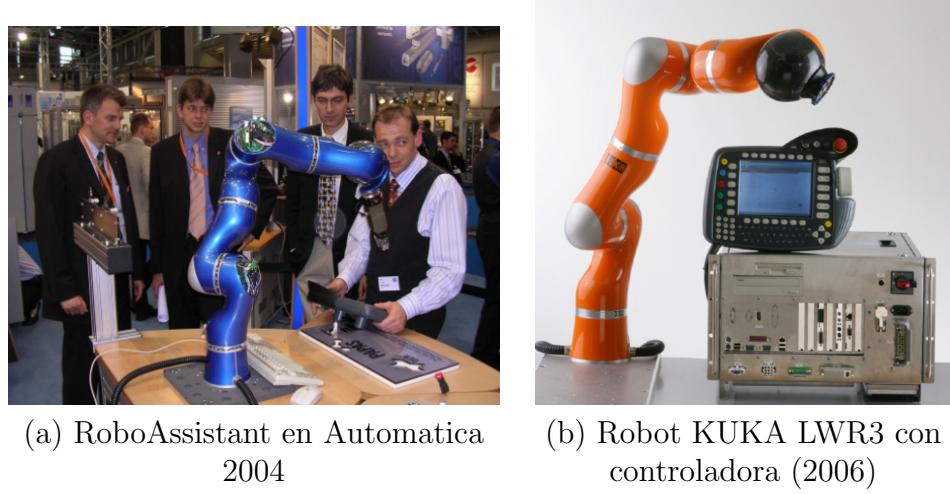


Figura 1.8: LWR3

A principios de 2007, dos estudiantes de doctorado de la Universidad de Standford, Keenan Wyrobek y Eric Bergerlas, pusieron las primeras piezas de lo que eventualmente se convertiría en ROS (Robot Operating System). Uno de los preceptos principales que se tuvo en cuenta para la creación de este sistema operativo para robots fue el de

crear un sistema que permitiese al máximo posible la reutilización de código, dando soporte a distintos tipos de robots y de aplicaciones. Esto resultó en la incorporación de ROS en una sorprendentemente amplia variedad de robots, extendiéndose incluso a dominios más allá de la comunidad académica de investigación a la que se dirigió inicialmente. Los años siguientes superaron todas las expectativas debido a que los avances en el ámbito de la robótica se compartieron de manera reproducible en ROS, y la Open Source Robotics Foundation (OSRF) se convirtió en el administrador principal de ROS en 2014. Con el objetivo de atender de manera más efectiva las demandas de una comunidad ROS más extensa y abordar sus nuevos escenarios de aplicación, la OSRF se dedicó a desarrollar ROS2 como un conjunto de paquetes paralelos que pudieran ser instalados junto a ROS1 (la versión original de ROS que nació en el año 2010)(Figura 1.9) y ser compatibles entre sí. Además, la popularidad de ROS ha seguido creciendo en la industria con el apoyo de proyectos como ROS-Industrial (ROS-I), una iniciativa de código abierto que extiende las capacidades avanzadas de ROS a hardware y aplicaciones industriales relevantes. [Suárez et al., 2022].

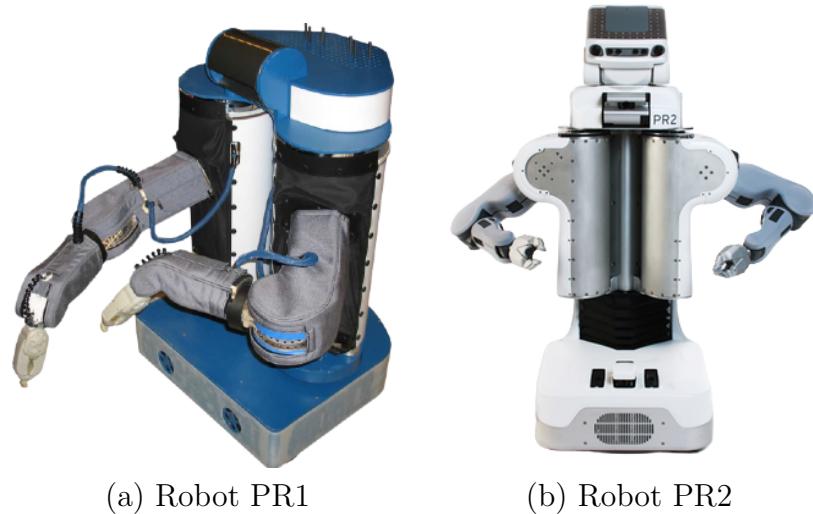


Figura 1.9: Robots utilizados para el desarrollo de ROS



Figura 1.10: Logo de ROS

En el año 2008 se entrega el primer robot colaborativo o cobot, el UR5 de Universal Robots<sup>3</sup> (ver Figura 1.11), considerado como uno de los logros tecnológicos más significativos de la década en la comunidad robótica. El brazo robótico es pionero en la programación 3D fácil de usar pero sofisticada, con una interfaz de usuario intuitiva que permite a cualquier persona configurarlo y utilizarlo de forma rápida. Esta empresa, fundada en el año 2005 por Esben Østergaard, Kasper Støy y Kristian Kassow tras conocerse en la Universidad de Dinamarca, surgió con el objetivo de hacer que la robótica sea accesible para las pequeñas y medianas empresas<sup>4</sup>.

Esben H. Østergaard, Director de Tecnología y cofundador de Universal Robots, tomó el trabajo original de Peskin y Colgate, dos investigadores de la empresa automovilística Ford de los años 90, que decidieron crear un nuevo robot industrial, más pequeño y ágil que los tradicionales, que saliera de su jaula para colaborar estrechamente con el ser humano en las tareas de calidad y personalización de los productos, sin embargo, no fueron capaces, puesto que el problema estaba en la relación entre seguridad y rendimiento, ya que el aumento de la primera reducía el de la segunda. Østergaard consiguió diseñar un sistema de seguridad y control para el cobot que lo bloquea en caso de colisión con el operario. Como recuerda el propio Østergaard, la seguridad era la clave con la que la robótica colaborativa podía entrar en el escenario industrial. Así se creó un robot capaz de operar en espacios confinados, en estrecho contacto con humanos, y sin instalar costosas barreras de seguridad [Cusano, 2022].

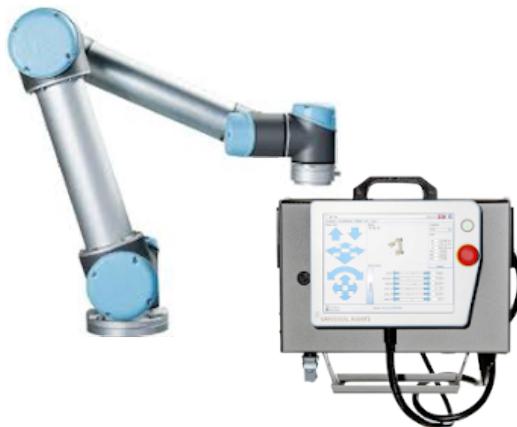


Figura 1.11: UR5 con su controladora

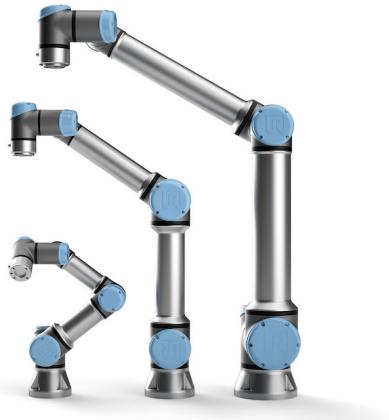
<sup>3</sup><https://www.universal-robots.com/es/>

<sup>4</sup><https://www.universal-robots.com/es/acerca-de-universal-robots/nuestra-historia/>

Más tarde, en 2018, Universal Robots presenta los robots colaborativos e-Series, que se pueden ver en la Figura 1.12, que incluían avances tecnológicos que permitían un desarrollo más rápido para una mayor variedad de aplicaciones, ofrecía una programación más sencilla y seguía las normas de seguridad ISO más actuales y recientes<sup>5</sup>.



(a) UR presenta los nuevos e-Series en Automatica 2018



(b) UR e-Series

Figura 1.12: Universal Robots e-Series

Esta cuarta generación de robótica industrial ha establecido un sólido punto de partida para una continua revolución en el campo de la automatización. Es esencial destacar que varios de los modelos de robots mencionados previamente han seguido evolucionando y mejorando con el tiempo, siendo fruto de estas mejoras, la comercialización de nuevos modelos y series. Debido a que la tecnología se encuentra en constante desarrollo y a la colaboración cada vez más estrecha entre humanos y robots, el futuro de la robótica industrial promete seguir transformando radicalmente nuestros métodos de trabajo y producción, abriendo así nuevas oportunidades y desafiando constantemente los límites de lo que podemos lograr en la automatización industrial, así como en los otros dos grandes grupos de la robótica, como la robótica de servicio y la robótica médica.

<sup>5</sup><https://www.universal-robots.com/es/acerca-de-universal-robots/nuestra-historia/>

### 1.1.2. Robots de servicio

Se define robot de servicio como un robot que realiza tareas útiles para las personas o los equipos, incluyendo en esta la manipulación o el servicio de artículos, el transporte, el apoyo físico, la orientación o información, el aseo personal, la cocina y la manipulación de alimentos y la limpieza en el ámbito personal; y la inspección, vigilancia, manipulación de objetos, transporte de personas, orientación o información, cocina y manipulación de alimentos y limpieza en el ámbito profesional [ISO/TC299, 2021].

Su historia se remonta a la década de 1960, cuando surgieron los primeros intentos de crear robots para ayudar en tareas domésticas y de atención al cliente. Uno de los precursores de estos robots de servicio en 1968 fue el robot Shakey, desarrollado por el Laboratorio de Investigación de Inteligencia Artificial de Stanford (Stanford Research Institute, SRI). Shakey (ver en Figura 1.13), provisto de múltiples sensores y medios para desplazarse por el suelo, además de control remoto por radio [Sánchez Martín et al., 2007b], podía realizar tareas de planificación, búsqueda de rutas y reordenación de objetos sencillos, siendo el primer robot móvil con capacidad para percibir y razonar sobre su entorno <sup>6</sup>.



Figura 1.13: Robot Shakey

---

<sup>6</sup><https://www.sri.com/hoi/shakey-the-robot/>

Sin embargo, fue a mediados de los años 80 cuando en los laboratorios y centros de investigación dedicados a la robótica se trató de revitalizar la importancia de los robots en nuestra sociedad, planteando las ventajas que el uso del robot podía traer en tareas en las que el ser humano asumía riesgos o en las que las capacidades de aquel estaban limitadas por factores como la fuerza o la precisión necesaria. Fue precisamente al entenderse que estas nuevas aplicaciones de la robótica no tenían un uso industrial con el objetivo de fabricar bienes, sino que se trataba de un empleo para desarrollar tareas para las personas, cuando se catalogaron como aplicaciones en el sector servicios [Barrientos, 2002] (ver ejemplos en la Figura 1.14).

En la práctica, las actuales y potenciales aplicaciones no industriales de los robots son tan variadas y diferentes que se dificulta su catalogación [Barrientos, 2002]; sin embargo, existen ciertas características especiales en estos robots de servicio que los hacen diferentes de los robots industriales [Aracil et al., 2008], y los caracterizan para llevar a cabo estas tareas para las personas, siendo las principales características estos tres atributos de diseño: representación, antropomorfismo y orientación a la tarea, es decir, los robots de servicio pueden tener una representación física (por ejemplo, Pepper) o tener una representación únicamente virtual (por ejemplo, Alexa, ya que el software de IA virtual que funciona de forma autónoma y aprende con el tiempo también puede clasificarse como robot de servicio), diseñarse como humanoides (es decir, antropomorfos) simulando una apariencia humana o como no humanoides (por ejemplo, el robot de limpieza Roomba de iRobot<sup>7</sup> (Figura 1.15), y pueden realizar tareas cognitivo-analíticas gracias a la potencia informática subyacente o tareas emocionales-sociales (por ejemplo, robots de recepción) [Wirtz et al., 2018].

---

<sup>7</sup><https://www.irobot.es/>

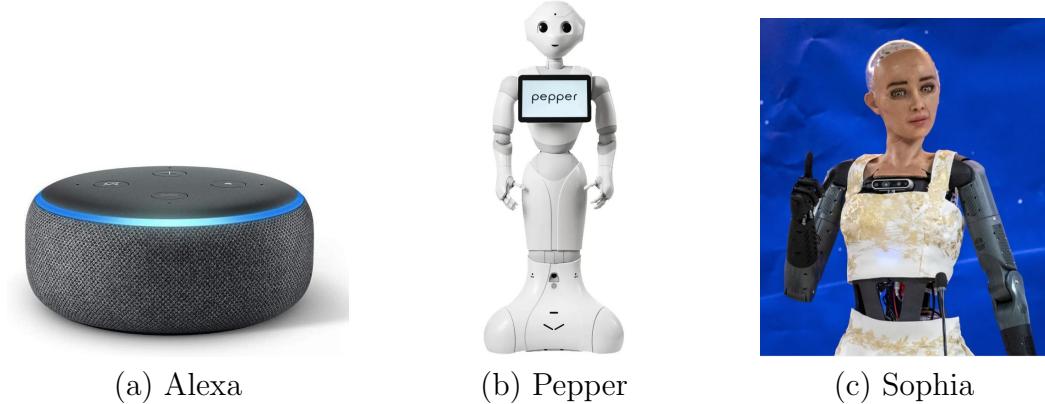


Figura 1.14: Robots de servicio

Tratando de establecer una división de los robots de servicio, la norma ISO 8373:2012, así como la Federación Internacional de Robótica o IFR, propuso clasificarlos en diferentes categorías según su función y aplicación en robots para uso doméstico y personal y robots de servicio destinados a un uso profesional [Gonzalez-Aguirre et al., 2021], siendo las aplicaciones más importantes las siguientes:

- *Limpieza:* Suelen estar equipados con sensores y tecnología de navegación que les permite moverse de manera autónoma por el espacio, detectar obstáculos y llevar a cabo actividades de limpieza de manera eficiente.



Figura 1.15: Robot aspirador Roomba de iRobot

- *Inspección y mantenimiento:* Son máquinas diseñadas para llevar a cabo tareas de supervisión, evaluación y mantenimiento en entornos de infraestructura o áreas de difícil acceso. Estos robots suele ser máquinas autónomas o teleoperadas equipadas con sensores, cámaras y herramientas especializadas que les permiten evaluar, reparar y mantener equipos, estructuras y sistemas en entornos desafiantes o peligrosos.



(a) Spot de Boston Dynamics

(b) ROBTET, robot para el mantenimiento de líneas de alta tensión

Figura 1.16: Robots de inspección y mantenimiento

- *Educación:* Aquellos robots utilizados en educación son robots diseñados para facilitar el aprendizaje y la enseñanza en los diferentes niveles educativos. Estos robots pueden ser utilizados en aulas, bibliotecas y entornos de aprendizaje para ayudar a los estudiantes a adquirir habilidades, fomentar la creatividad y brindar experiencias educativas interactivas.



(a) Robot NAO

(b) Turtlebot 4

Figura 1.17: Robots de educación

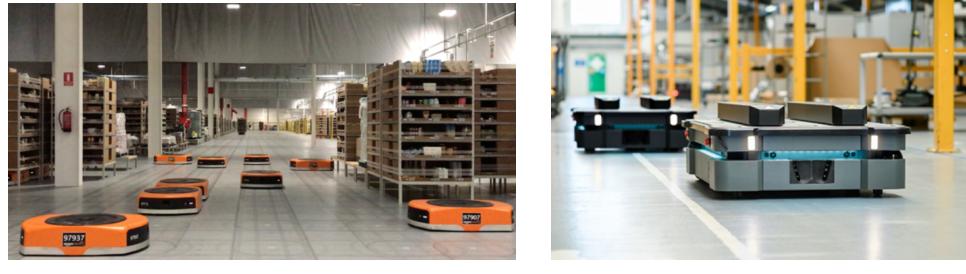
- *Logística:* Los robots de servicio utilizados en logística son robots diseñados para llevar a cabo tareas relacionadas con la gestión y el movimiento de mercancías y productos en entornos de almacenamiento, distribución y transporte. Estos robots desempeñan un papel fundamental en la optimización de la cadena de suministro, mejorando la eficiencia y la precisión en la manipulación de productos.

Un ejemplo del posible uso de estos robots en el sector agrícola, es la primera granja vertical de interior del mundo en Estados Unidos, que producirá 18 millones de kilogramos de fresas al año, marcando un hito en la agricultura moderna, y demostrando que la automatización e integración de la robótica en este tipo de granjas verticales puede transformar la producción y recolección de alimentos a gran escala [EcoInventos.com, 2024].



Figura 1.18: Agricultura vertical

Por otro lado, dentro de los robots de logística empleados para el movimiento de mercancías, se pueden distinguir dos grandes grupos: Vehículos Guiados Automáticos (AGV) y Robots Móviles Autónomos (AMR). La principal diferencia entre los AMR y los AGV radica en la capacidad de adaptación y autonomía, ya que los AMR son altamente flexibles, autónomos y versátiles, con capacidades autónomas avanzadas que les permiten ajustarse rápidamente a cambios en los entornos de trabajo, mientras que los AGV siguen rutas predefinidas y son adecuados para tareas de transporte predecibles en entornos más controlados.



(a) AGV Robots Kiva de Amazon

(b) AMRs de MiR

Figura 1.19: Robots de logística

- *Entretenimiento:* Son robots diseñados específicamente para proporcionar experiencias lúdicas y de entretenimiento a las personas. Estos robots se utilizan en una variedad de contextos, siendo máquinas robóticas diseñadas para interactuar con el público.



(a) SONY Aibo



(b) Dron DJI Spark

Figura 1.20: Robots de entretenimiento

La robótica de servicio representa una revolución en la asistencia y el apoyo a diversas industrias, desde la logística hasta la atención al cliente en el comercio minorista. Sin embargo, su impacto va más allá, extendiéndose hasta la atención médica. En este contexto, la robótica médica emerge como una vanguardia tecnológica que fusiona la innovación robótica con la medicina moderna para ofrecer soluciones innovadoras en diagnóstico, tratamiento y rehabilitación, demostrando su potencial para revolucionar la forma en que brindamos y recibimos atención médica.

### 1.1.3. Robot Médico

Se define *robot médico* como aquellos dispositivos electromecánicos que desempeñan parcial o totalmente algunas funciones de los seres humanos o de sus órganos al resolver problemas médicos, ayudando a mejorar la asistencia al paciente y los resultados, a la vez que aumenta la eficiencia operativa [Kraevsky and Rogatkin, 2010].

Los robots médicos se desarrollaron por primera vez hace poco más de tres décadas para permitir a los cirujanos operar a sus pacientes a distancia o con mayor precisión. A finales de los años noventa, había 2 tipos de telemanipuladores quirúrgicos aprobados por la Administración de Alimentos y Medicamentos de los Estados Unidos (FDA): el Zeus y el da Vinci (Figura 1.21), introducido en 1998-1999, que permitía aumentar la precisión de las cirugías mínimamente invasivas (CMI) [Romero-Tamarit et al., 2020].



Figura 1.21: Robot Da Vinci

Las primeras aplicaciones fueron en los campos de neurocirugía y cirugía ortopédica, siendo la cirugía donde mayor impacto han tenido los robots médicos. A medida que se ha hecho patente la aceptación de los robots quirúrgicos por nuestros sistemas sanitarios, los investigadores en robótica han ido centrando cada vez más su atención en cómo podría ser la próxima generación de robots médicos. Su atención no se limita a los robots quirúrgicos, y también se están investigando otras áreas de la medicina, como los robots para realizar rehabilitación física con pacientes con discapacidades motores, como el exoesqueleto Ekso Bionics, robots de telepresencia para la interacción del paciente con el personal sanitario externo, como el robot RP-VITA, automatización de farmacias, robots para desinfectar clínicas, etc. [Dupont et al., 2021]

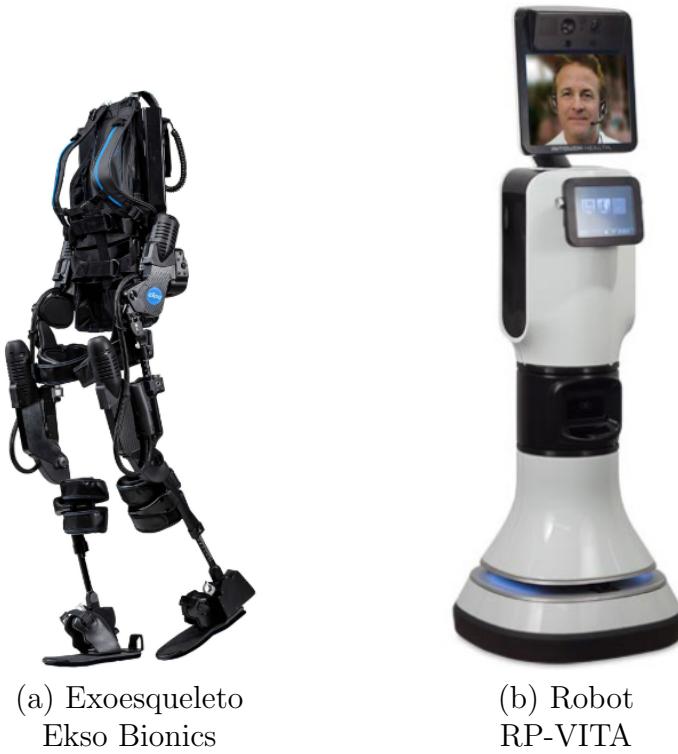


Figura 1.22: Robots médicos

El rápido crecimiento de la robótica médica se debe a una combinación de mejoras tecnológicas ( motores, materiales y teoría de control), los avances en imagen médica (mayor resolución, resonancia magnética y ecografía 3D) y una mayor aceptación por cirujanos y pacientes de los procedimientos laparoscópicos y la asistencia robótica [Beasley, 2012], convirtiéndose en un campo interdisciplinario que abarca desde cirugía asistida por robots hasta sistemas de diagnóstico de vanguardia. Gran parte de su éxito radica en la integración de tecnologías avanzadas, como la inteligencia y la visión artificial. Estas disciplinas están redefiniendo la forma en que los robots médicos pueden interactuar con el entorno, interpretar datos y, en última instancia, mejorar los resultados en la atención médica. A continuación, explicaremos el impacto que la inteligencia y la visión artificial están teniendo en la robótica, y las capacidades y oportunidades que estas presentan en una inmensa variedad de aplicaciones.

## 1.2. Inteligencia Artificial

La Inteligencia Artificial (IA) es un área de la ciencia de gran interés por ser un área multidisciplinaria donde se realizan sistemas que tratan de hacer tareas y resolver problemas como lo hace un humano; así mismo, trata de simular de manera artificial las formas de pensamiento y de trabajar del cerebro para la toma de decisiones [Ponce Gallegos et al., 2014].

El origen del concepto y de los criterios de desarrollo de la IA se remontan al año 1936, con el matemático inglés Alan Turing, quien definió una máquina abstracta, la Máquina de Turing, un modelo matemático que consiste en un autómata capaz de implementar cualquier problema matemático expresado por medio de un algoritmo, que sirvió de base de la noción de algoritmo y la definición de clase de problemas deducibles [Hardy, 2001], y quien intuyó la importancia que jugaría el aprendizaje automático en el desarrollo de la IA al afirmar que, en lugar de intentar emular mediante una máquina la mente de un adulto, quizás sería más factible intentar emular la mente de un niño y luego someter a la máquina a un proceso de aprendizaje que diera lugar a un desarrollo cognitivo de dicha mente hasta alcanzar el equivalente de una mente adulta, lo que actualmente se conoce como robótica de desarrollo [González and de Mántaras Badia, 2017], mientras que el apelativo Inteligencia Artificial se debe a John McCarthy, quien organizó una conferencia en el Dartmouth College (Estados Unidos) en agosto de 1956, para discutir sobre la posibilidad de construir máquinas “inteligentes”. Como resultado de esta reunión, se establecieron las primeras bases sobre la inteligencia de los computadores [Ponce Gallegos et al., 2014].

La historia de la IA ha sido testigo de ciclos caracterizados por la introducción de nuevos y creativos enfoques y de un sistemático perfeccionamiento de los mejores. Estos ciclos de avance y desafío continúan moldeando su evolución y prometen un futuro emocionante y lleno de posibilidades.

Dentro de las diversas formas de clasificar la IA, existe una clasificación, como se muestra en la Figura 1.23), que se basa en el objetivo y la forma en que trabaja el sistema: sistemas que piensan como humanos, sistemas que actúan como humanos, sistemas que piensan racionalmente, y sistemas actuantes racionales. Esta clasificación de manera inicial se veía como clases independientes, sin embargo, en la actualidad los sistemas mezclan características de ellas. [Ponce Gallegos et al., 2014]



Figura 1.23: Modelos de inteligencia

La clasificación de los modelos de IA tiene un impacto significativo en las aplicaciones de la IA, ya que determina cómo se desarrollan y aplican los sistemas de inteligencia artificial en una amplia gama de campos, contribuyendo así a la mejora de la eficiencia y la innovación en muchas industrias donde se han ido desarrollando diferentes herramientas propias y aplicaciones, dando lugar a multitud de ramas que parten de la IA. Una de las ramas más fascinantes y prometedoras de la inteligencia artificial es la visión artificial, que busca dotar a las máquinas de la capacidad de interpretar y comprender el mundo visual que les rodea. Esta sección del capítulo de introducción se centra en la importancia de la IA y su intersección con la visión artificial, explorando cómo estas disciplinas se fusionan para mejorar la percepción y la comprensión de imágenes y vídeos.

### 1.3. Visión Artificial

La visión artificial se define como la ciencia de programar un ordenador para procesar imágenes o vídeos e incluso entenderlos [Culjak et al., 2012].

En [Bradski and Kaehler, 2008] explica cómo es la transformación de datos desde un fotograma o vídeo cámara hasta lo que puede ser una decisión o una nueva representación [Alvear-Puertas et al., 2017]. Para ello, la imagen percibida pasa por los procesos de obtención, caracterización e interpretación de información de imágenes, y estos procesos, pueden ser subdivididos a su vez en [García Santillán and Caranqui Sánchez, 2015]:

Procesos	Nivel de Visión	Entrada	Salida	Área
1. Captura 2. Pre-procesamiento	Bajo	Imagen	Imagen	Procesamiento de imágenes
3. Segmentación	Medio	Imagen	Grupo de píxeles en bruto (objetos o regiones)	Análisis de Imágenes
4. Descripción		Objetos o regiones	Información cuantitativa de los objetos o regiones	
5. Reconocimiento (clasificación)		Información cuantitativa	Objetos clasificados en categorías	
6. Interpretación	Alto	Objetos clasificados en categorías	Compresión de la escena	Visión por Computador

Cuadro 1.1: Procesos de la visión artificial

1. *Captura*: La captura o digitalización es el proceso en el que se obtiene una imagen digital a partir de una imagen analógica a través de un dispositivo para que pueda ser manipulada por un ordenador. Esta imagen estará representada como una matriz de números (píxeles) [Martínez Madruga, 2022].
2. *Pre-procesamiento*: En esta fase, se incorporan métodos destinados a restaurar las imágenes capturadas, dado que es factible que las imágenes experimenten deterioro, como la disminución de su claridad o la presencia de ruido. Esta etapa tiene como objetivo corregir estos problemas mediante procedimientos como la eliminación de ruido o la mejora del contraste y la nitidez.

3. *Segmentación:* La segmentación consiste en dividir una imagen en regiones o componentes más pequeños (grupo de píxeles) con el objetivo de identificar y aislar objetos o áreas de interés dentro de la imagen. El propósito principal es simplificar y organizar la información visual contenida en la imagen para que sea más fácil de analizar, comprender y procesar por ordenador.
4. *Descripción:* Es el proceso que obtiene características relevantes para poder diferenciar un tipo de objeto de otro, pudiendo ser externas, como la forma, el perímetro o el rectángulo mínimo que contiene la región; o internas, como el área o el centro de gravedad, entre otros [García Santillán and Caranqui Sánchez, 2015].
5. *Reconocimiento (clasificación):* Esta fase se centra en identificar y asignar etiquetas o categorías a objetos o patrones previamente segmentados en una imagen o secuencia de imágenes. El proceso de reconocimiento implica el uso de algoritmos y técnicas de aprendizaje automático, como redes neuronales artificiales o métodos estadísticos, entre otros, para entrenar un modelo que pueda tomar las características extraídas y realizar predicciones sobre la clase o categoría a la que pertenecen los objetos detectados.
6. *Interpretación:* Esta fase implica el análisis y comprensión del significado y el contexto de la información visual obtenida de las imágenes o la secuencia de imágenes capturadas. Esta etapa implica razonamiento, toma de decisiones y puede requerir el procesamiento de lenguaje natural para obtener una comprensión más profunda del contenido visual.

Estas fases son las empleadas bajo el paradigma de lo que se conoce como Visión Artificial Clásica, enfocada a la utilización de algoritmos específicos para procesar imágenes y reconocer en ellas características básicas [Martínez Madruga, 2022], siendo generalmente secuenciales, a pesar de que los procesos que se utilicen en la resolución de un determinado problema dependen de su complejidad y no todas pueden ser siempre necesarias [García Santillán and Caranqui Sánchez, 2015]. Sin embargo, para mejorar aún más la eficacia de los sistemas de visión artificial, se recurre al aprendizaje automático o *machine learning* (ML). A continuación, profundizaremos en el papel del *machine learning* en la visión artificial y su importancia en la creación de sistemas inteligentes de procesamiento de imágenes.

## 1.4. Machine Learning

El Machine Learning (Aprendizaje Automático) es una rama en evolución de la Inteligencia Artificial que se encarga de generar algoritmos que tienen la capacidad de aprender del entorno circundante y no tener que programarlos de manera explícita, teniendo en cuenta todos los escenarios posibles, a partir de la construcción de modelos analíticos [Sandoval Serrano et al., 2018] cuyos aspectos clave y semántica se muestran en la Figura 1.24.

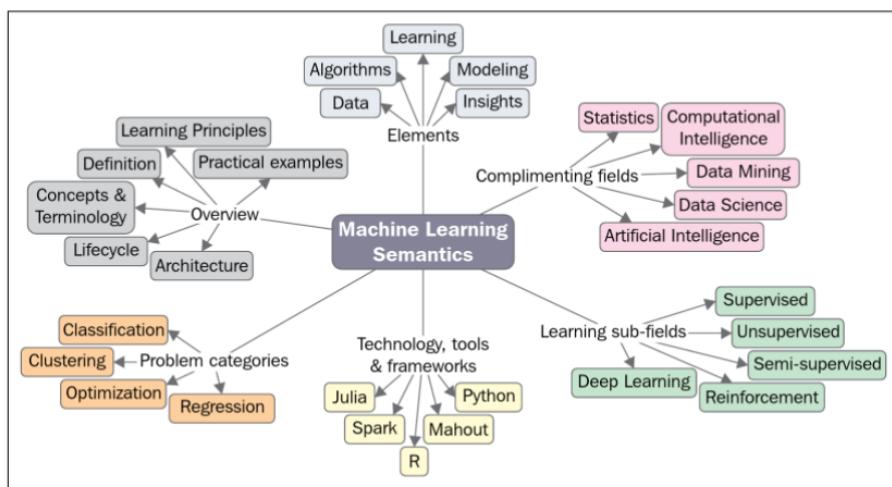


Figura 1.24: Aspectos clave y semántica del Aprendizaje automático

En función del problema planteado y de los datos disponibles, se pueden distinguir tres tipos de ML:

- *Aprendizaje supervisado*: Consiste en operar a partir de una expectativa conocida, en este contexto, los conjuntos de datos de entrada también se denominan conjuntos de datos etiquetados. Los algoritmos clasificados en esta categoría se centran en establecer una relación entre los atributos de entrada y salida, y utilizan esta relación de forma especulativa para generar una salida para nuevos puntos de datos de entrada [Gollapudi, 2016].
- *Aprendizaje no supervisado*: El análisis o aprendizaje no supervisado se basa en el análisis de clasificación en el que no empezamos con un objetivo específico en mente, también denominado agrupación. En este caso, el objetivo es descifrar la estructura de los datos a partir de la construcción de un mapa entre los atributos de entrada y salida y, de hecho, los atributos de salida no están definidos. Estos algoritmos de aprendizaje operan sobre un conjunto de datos no etiquetados por esta razón [Gollapudi, 2016].

- *Aprendizaje por refuerzo*: En lugar de proporcionar pares de entrada y salida, se describe el estado actual del sistema, se especifica un objetivo proporcionando una lista de acciones permitidas y sus restricciones ambientales para sus resultados, y se deja que el modelo de ML experimente por sí mismo el proceso para alcanzar el objetivo utilizando el principio de ensayo y error para maximizar la recompensa del resultado [Janiesch et al., 2021].

Del mismo modo, dependiendo de la tarea de aprendizaje, existen varias clases de algoritmos de ML, cada uno de ellos con múltiples especificaciones y variantes, que pueden englobarse o bien en el Shallow Machine Learning (aprendizaje superficial), que se centra en algoritmos más simples para realizar tareas específicas, o en Deep Learning (aprendizaje profundo), que utiliza la construcción y entrenamiento de Redes Neuronales Artificiales (RNA), un tipo de modelo inspirado en la estructura y funcionamiento del cerebro humano, tal y como se puede apreciar en el diagrama de la Figura 1.25 [Janiesch et al., 2021].



Figura 1.25: Diagrama de Venn de la relación entre distintas áreas de la IA

El Machine Learning, abarca desde enfoques más superficiales hasta técnicas más avanzadas, tal y como se ha podido observar, sin embargo, es el Deep Learning lo que realmente potencia la capacidad de las máquinas para aprender y generalizar patrones complejos de manera excepcional. En la próxima sección, se hablará sobre el Deep Learning, centrándose en las RNA y en cómo estas posibilitan abordar tareas más complejas, como el reconocimiento de patrones en imágenes o el procesamiento de lenguaje natural.

## 1.5. Deep Learning

El Deep Learning o aprendizaje profundo, constituye una rama de la IA, incluida dentro del Machine Learning, cuyos modelos computacionales se inspiran en el funcionamiento del cerebro humano y se diseñan con el propósito de adquirir conocimientos y llevar a cabo tareas específicas mediante el procesamiento de datos. La etiqueta *profundo* hace referencia a la presencia de múltiples capas de neuronas artificiales en la red, lo que facilita la representación jerárquica de características y potencia la capacidad para aprender y poder diferenciar patrones complejos a partir de los datos.

Las Redes Neuronales Artificiales (RNA) o Artificial Neural Networks (ANN) en inglés, están inspiradas en las redes neuronales biológicas del cerebro humano tal y como muestra la Figura 1.26, presentando características del mismo, ya que estas aprenden de la experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos. En las RNA, la unidad análoga a la neurona biológica es el elemento procesador, PE (Process Element). Un PE tiene varias entradas y las combina, normalmente, con una suma básica. La suma de las entradas es modificada por una función de transferencia y el valor de la salida de esta función de transferencia se pasa directamente a la salida del elemento procesador. Existen dos capas con conexiones con el mundo exterior, una capa de entrada o buffer de entrada, donde se presentan los datos a la red, y una capa o buffer de salida que mantiene la respuesta de la red a una entrada, mientras que el resto de las capas reciben el nombre de capas ocultas [Basogain, 2008].

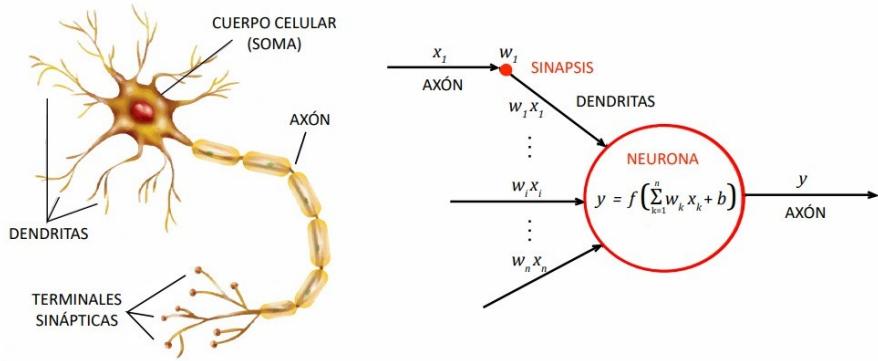


Figura 1.26: Modelo biológico de una neurona genérica, comúnmente una neurona motora típica (izquierda) y el respectivo modelo matemático (derecha)

En consecuencia, se puede construir una red neuronal artificial mediante un conjunto de neuronas artificiales, es decir, mediante un conjunto de funciones, y conectando comúnmente la salida de cada una a las entradas de otras diferentes. De esta manera, las RNA no son más que redes de funciones, típicamente representadas mediante la composición de varias funciones como se representa en la Figura 1.27. Esto hace que los diferentes modelos de redes neuronales difieran principalmente en las funciones de activación utilizadas, el patrón de interconexión, e inclusive el tiempo de transmisión de la información. Es importante señalar que la característica clave de la sinapsis, el escalar las señales de entrada por factores (pesos), es la manera en la que se cree que el cerebro aprende. Por lo tanto, distintos pesos dan como resultado diferentes respuestas a una entrada. De esta manera, se puede decir que el aprendizaje es el ajuste de los pesos en respuesta a un estímulo [Dinamarca, 2018].

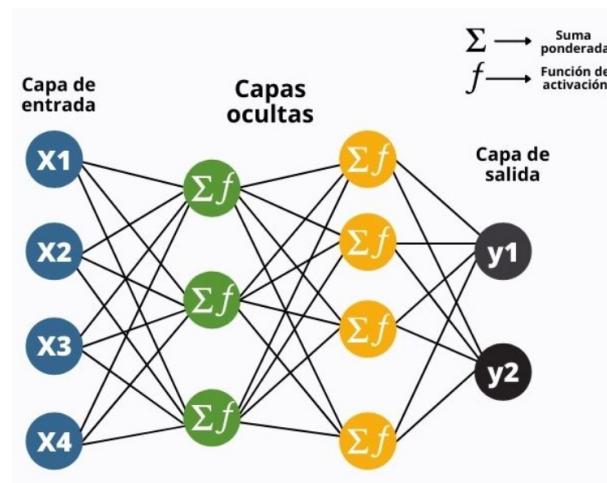


Figura 1.27: Arquitectura de una red neuronal

La historia de las redes neuronales se remonta a la década de 1940, cuando el neurofisiólogo Warren McCulloch y el matemático Walter Pitts modelaron una red neuronal simple utilizando circuitos eléctricos. Su modelo estaba basado en la idea de lógica de umbral, que más tarde sería fundamental para el desarrollo de las redes neuronales artificiales. En 1958, el neurobiólogo Frank Rosenblatt dio un paso significativo al comenzar a trabajar en el perceptrón, un modelo computacional que representaba la unidad básica de una red neuronal y realizaba una suma ponderada de varias entradas, al tratarse de una unidad de procesamiento, llevando a cabo una combinación lineal de esas entradas, y aplicando posteriormente una función de activación para producir una salida. Este trabajo marcó el inicio de la investigación en redes neuronales artificiales. Saltando a la década de 2010, dos factores contribuirían a la revolución de aplicaciones de redes neuronales y algoritmos de aprendizaje profundo. En primer lugar, los avances en hardware especializado aceleraron drásticamente el entrenamiento y el rendimiento de las redes neuronales, reduciendo su consumo de energía, mientras que, en segundo lugar, el aumento de datos abiertos disponibles en línea y servicios de bajo costo para etiquetar datos impulsaron el desarrollo de la inteligencia artificial [Abeliuk and Gutiérrez, 2021].

En este capítulo se ha introducido el nacimiento y la historia de los robots y la robótica tal y como los conocemos hoy en día, dentro de cuya rama encontramos uno de los tres grandes grupos en los cuales puede dividirse esta, la Robótica de Servicio, y para la que la Inteligencia Artificial, y más concretamente el campo de la Visión Artificial junto con el del Deep Learning, siendo este subcategoría del Machine Learning, juegan un papel fundamental en el desarrollo de nuevas aplicaciones.

En este proyecto se presenta un sistema que, mediante Visión Artificial y Machine Learning, es capaz de reconocer la maduración de frutos, más concretamente de fresas, con el objetivo de poder ayudar así a mejorar el proceso de recolección. En los siguientes capítulos de este trabajo se detallarán los objetivos del mismo, delineando claramente las metas; se expondrá la plataforma de desarrollo, detallando las herramientas seleccionadas para la elaboración del proyecto; se presentará el diseño y la arquitectura del proyecto; y, finalmente, llegaremos a las conclusiones, donde tendrá lugar una breve recopilación de información sobre los resultados obtenidos y las posibles direcciones futuras.

---

## Capítulo 2

# Estado del arte

---

En el presente capítulo, se van a describir algunos de los prototipos y soluciones más destacables aplicadas a la detección y recolección de fresas usando inteligencia artificial y técnicas robóticas.

En los sistemas de producción de cultivos, hay operaciones de campo que requieren mucha mano de obra, ya sea por su complejidad o por el hecho de que están relacionadas con una interacción sensible entre plantas y productos comestibles, o por la repetitividad que requieren a lo largo de un ciclo de producción de cultivos. Estos son los factores clave para el desarrollo de robots agrícola, tal y como se relata en el artículo [Fountas et al., 2020], donde se realiza una revisión sistemática de la bibliografía existente sobre la investigación y la robótica agrícola comercial utilizada en las operaciones de los campos de cultivo en función de las principales operaciones de campo, siendo estas: deshierbe, siembra, detección de enfermedades e insectos, exploración de cultivos (seguimiento de plantas y fenotipado), pulverización o rociado, cosecha, robots de gestión de plantas y sistemas robóticos polivalentes.



Figura 2.1: Representación de varios tipos de robots agrícolas

Entre estas operaciones se escogió centrarse en la recolección, que es una de las tareas más laboriosas y repetitivas, a la vez que forma parte de todos los ciclos de producción en la agricultura, existiendo dos tipos de cosechadoras robotizadas: a granel (se recogen todas las frutas/verduras) y selectivas (sólo se recogen las frutas maduras/listas para ser cosechadas) [Fountas et al., 2020], dentro de las cuales se engloba este trabajo final de grado.

La mayoría de los robots de recolección se centran en las fresas, un cultivo de alto valor, que sufre un alto coste de producción debido principalmente al coste de la mano de obra, sobre todo durante la recolección. Los robots de recolección de fresas más rápidos fueron los robots con velocidades de recolección de 7,5 y 8,6 segundos por fresa, respectivamente, y el robot Berry 5 de Harvest Croo<sup>8</sup>, con una velocidad de recolección declarada de 8 segundos por fruta.

Sin embargo, no basta con tener una alta velocidad de recogida, ya que también es importante una alta tasa de recogida, siendo la más alta presentada del 86 %. Para evaluar el rendimiento de los robots en la recolección, se ha mostrado un rango en las tasas de éxito de recolección, desde el 0 % hasta el 64 %, para varias categorías, tal y como se muestra en la ilustración de la Figura 2.2, mereciendo la pena mencionar que para la recolección de fresas, parece haber un equilibrio entre la velocidad y la tasa de recolección, ya que las velocidades de recolección rápidas van acompañadas de tasas de recolección bajas y viceversa [Fountas et al., 2020].

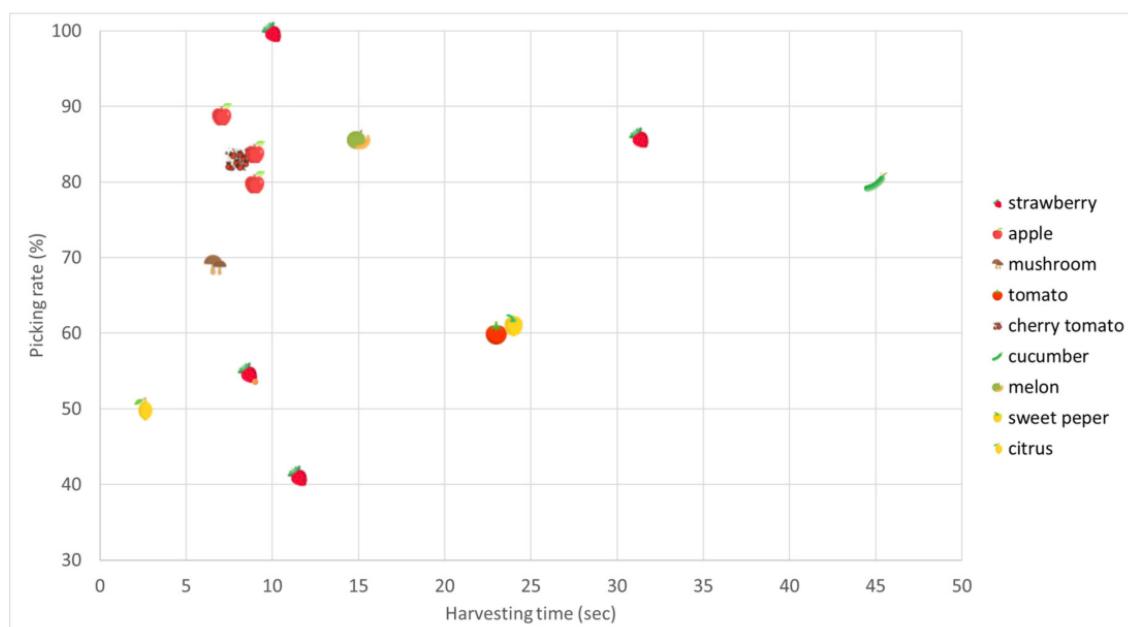


Figura 2.2: Ilustración global del rendimiento general de los robots revisados

<sup>8</sup><https://www.harvestcroorobotics.com/>

En 2009, después de más de 30 años de comercializar fresas en la provincia de Huelva, y de algunos intentos infructuosos, la empresa AGROBOT<sup>9</sup>, con sede en el Centro de Innovación y Tecnología que la Consejería de Innovación tiene en Lepe (Huelva), consiguió poner en marcha un prototipo con 40 brazos robóticos que identificaba los frutos maduros y los recogía sin dañarlos, siendo capaz de clasificarlas y colocarlas en los envases que recorren las cintas transportadoras, gracias a un grupo de ingenieros liderado por Juan Bravo [Cabanillas, 2009]. Tras la presentación del prototipo y las primeras pruebas con éxito, la empresa onubense desarrolló el prototipo final, el Agrobot SW 6010, una cosechadora de fresas que es capaz de recoger de la mata solo la fruta que está madura mediante inteligencia artificial con 30 brazos robóticos que incorporan una cámara con visión artificial que detecta el grado de madurez de la fruta. Recopila entre 10 y 30 imágenes por segundo, permitiendo a los algoritmos confirmar, mediante un análisis morfológico y de color en tiempo real, si la fresa cumple con los parámetros marcados de tamaño, grosor y color. Si la fresa está verde se quedará en la planta, siendo la robótica quien corta el tallo de la fruta y la deposita en una cinta transportadora para que posteriormente sean los humanos quienes clasifiquen las fresas y las guarden en las cajas para su distribución [Fernández, 2019]. Este modelo se volvería a mejorar para obtener el Agrobot E-Series<sup>10</sup> (Figura 2.3), que consta de hasta 24 brazos robóticos independientes, permitiendo adaptarse a cualquier configuración agrícola, y que, fabricado en acero inoxidable y aluminio de calidad militar, puede funcionar de forma robusta con un alto grado de precisión, ya que su arquitectura de brazos descentralizada y autosuficiente hace que sea especialmente fácil de configurar y mantener, y los sensores de profundidad infrarrojos y en color integrados de corto alcance a bordo, que sirven para captar todos los detalles están respaldados por unidades de procesamiento gráfico, que ayudan a evaluar el grado de madurez de la fruta, incorporando a su vez, sensores LiDAR que se encargan de la seguridad de los trabajadores del campo circundantes.

<sup>9</sup><https://www.agrobot.com/?lang=es>

<sup>10</sup><https://www.agrobot.com/e-series>

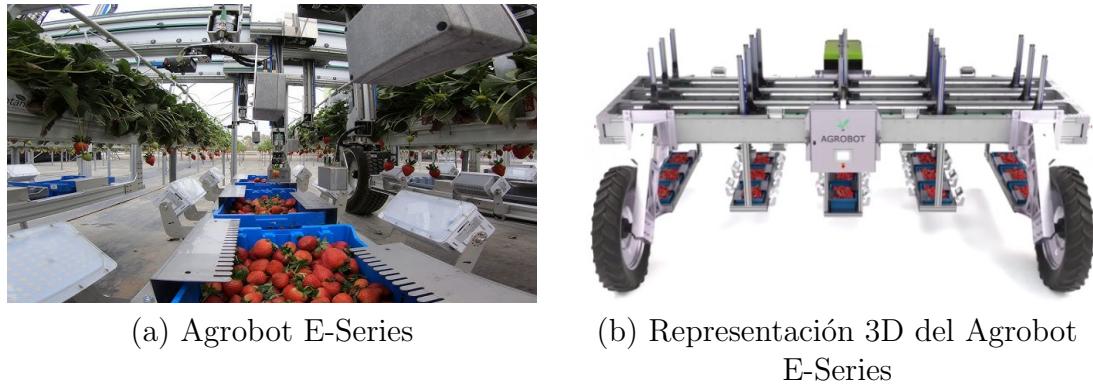


Figura 2.3: Agrobot

La empresa belga de I+D agrícola Octinion<sup>11</sup> desarrolló un prototipo de robot recolector de fresas en 2017, que recoge los frutos de forma totalmente autónoma basándose en el método de cultivo habitual (sobre mesa), con el fin de resolver el obstáculo emergente de la agricultura occidental: la falta de mano de obra asequible que pone en peligro la sostenibilidad y conservación del negocio [De Preter et al., 2018].

Este robot, cuyo diseño está representado en la Figura 2.4, está formado por un vehículo eléctrico consistente en una plataforma eléctrica con una batería recargable; un sistema de localización constituido por codificadores de rueda, un giroscopio y un sistema de posicionamiento en interiores de banda ultra ancha (UWB); tres cámaras RGB utilizadas para la detección de las fresas por cámara mediante visión artificial, un brazo robótico diseñado a medida, la pinza que se acopla al extremo del brazo robótico y agarra con sus dedos la fresa detectada, un módulo de gestión o manipulación logística consistente en varias cestas que se transportan en la plataforma eléctrica y que están preparadas por el robot para que estén inmediatamente listas para el envasado final y el transporte; y un módulo de control de calidad que clasifica las fresas detectadas en función de su madurez, forma, tamaño y dulzor [De Preter et al., 2018].

---

<sup>11</sup><http://octinion.com/products/agricultural-robotics/rubion>

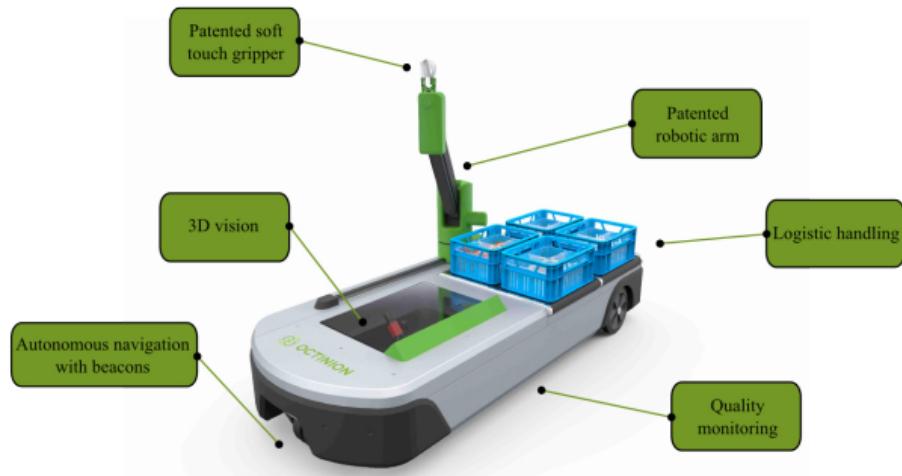


Figura 2.4: Diseño conceptual del robot de recogida con sus componentes

Todo esto le permite al robot recoger al menos el 70 % de las fresas maduras, siempre sin dañarlas, ya que sólo decide recoger la fruta si su acción no va a dañar otras fresas, siendo el tiempo necesario para desplazarse hasta la fresa, cogerla y depositarla en una cesta (caja en la que se colocan las fresas) de 4 segundos, siendo la calidad y velocidad de recolección comparables a las de un recolector humano ideal [De Preter et al., 2018]. Además de esto, el uso de este robot para cosechar las fresas de forma autónoma, presenta algunas ventajas adicionales:

- Mayor calidad de recolección (disminución de los daños en la fruta).
- Mayor calidad de clasificación (más categorías, colocación más óptima en los envases, categorización mejor y más flexible).
- La productividad del robot es constante y predecible. No necesita formación, por lo que no supone un coste elevado al principio.
- Nuevas herramientas de gestión con los datos recopilados: predicción del rendimiento, asignaciones de cosecha más complejas (recogida de un tamaño determinado, recogida un día antes o después).
- Horario de recogida ilimitado (fines de semana, horario nocturno).

El sistema Dogtooth<sup>12</sup> de Cambridge (Reino Unido) (ver en Figura 2.5) fue desarrollado para mejorar las operaciones de recogida de frutos rojos siendo capaz de navegar por hileras de fresas y frambuesas, detectar y localizar las maduras, así como recoger y comprobar la fruta antes de colocarla en un cesto, permite a las empresas de recolección de fruta sustituir el método de recogida manual y ahorrar tiempo. Sin embargo, al cortar el tallo produce una pequeña herida, que permite que las enfermedades entren fácilmente en la planta, y la parte restante del pedúnculo puede magullar otras fresas durante el transporte, cuando las frutas se agitan en sus cajas. Así mismo, Dogtooth parte de un robot industrial caro cuya cinemática no está optimizada para la recogida de fresas, suponiendo un inconveniente frente a sus competidores que, a pesar de encontrarse en fase de prototipo, sus conceptos exigen cambios drásticos en la infraestructura ya que las fresas recolectadas no cumplen las especificaciones exigidas por el mercado.



(a) Robot Dogtooth



(b) Brazo robótico del Dogtooth

Figura 2.5: Robot agrícola Dogtooth

---

<sup>12</sup><https://dogtooth.tech/>

El sistema explicado en [Xiong et al., 2019], presenta el desarrollo y la evaluación de un robot para la recolección de fresas (*Fragaria × ananassa*) cultivadas en invernaderos (ver Figura 2.6). El robot encargado de realizar esta tarea, está compuesto por una pinza montada en un brazo industrial que a su vez está montado en una base móvil junto con una cámara RGB-D, que utilizará el sistema de visión que se basa en el umbral de color combinado con el cribado del área del objeto y el rango de profundidad para seleccionar las fresas maduras y alcanzables. La novedosa pinza está diseñada para apuntar a la fruta y no al tallo, por lo que sólo requiere la ubicación de la fruta para la recolección. Además, está equipada con sensores internos, por lo que la pinza puede detectar y corregir errores de posición, y es resistente a los errores de localización introducidos por el módulo de visión. Otra característica importante de la pinza es el contenedor interno que se utiliza para recoger las bayas durante la recolección, lo cual reduce considerablemente el tiempo de recogida dado que el manipulador no tiene que ir y venir entre cada fresa a una cesta separada.

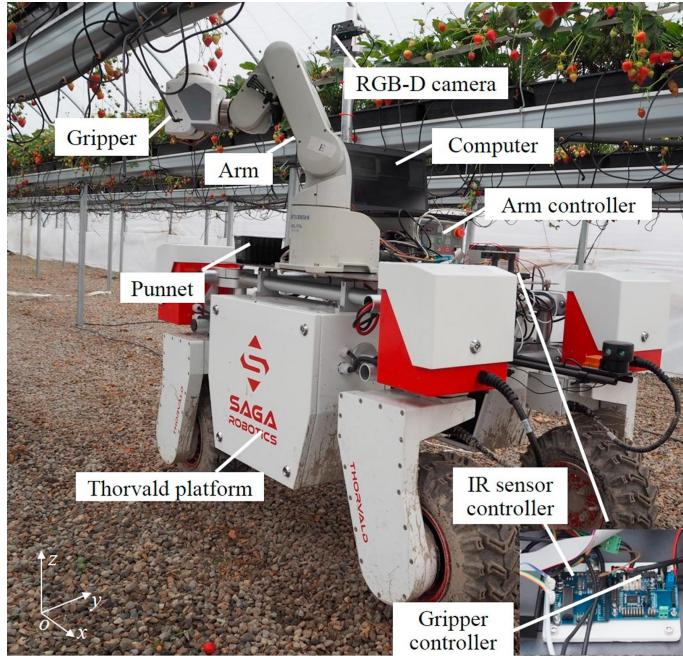


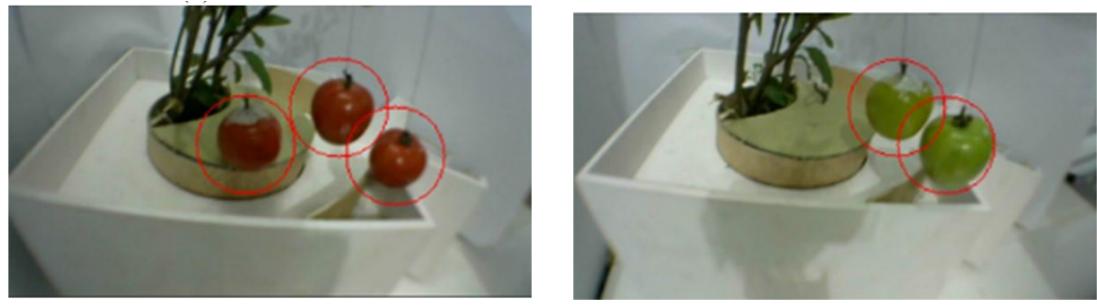
Figura 2.6: Montaje del hardware en una explotación de fresas: el robot consta principalmente de una cámara RGB-D, un brazo 5-DOF, una pinza y una plataforma móvil

Con todo esto, los experimentos de campo muestran la duración media del ciclo de recolección continua de una sola fresa es de 7,5 s y de 10,6 s cuando se incluyen todos los procedimientos. Además, el robot es capaz de recoger fresas aisladas con una tasa de éxito cercana a la perfección (96,8 %). Sin embargo, en las explotaciones agrícolas, el porcentaje medio de éxito es del 53,6 %, y del 59,0 % si se incluye el «éxito con daños».

Este descenso de la tasa de éxito es debido a los siguientes factores:

- Oclusión de las fresas, lo que deriva en una detección fallida y una no recolección de las mismas.
- Posibles detecciones duplicadas debido a las agrupaciones de fresas que se tocan entre sí.
- Errores de localización demasiado grandes para la que la pinza pueda coger la fresa producidos por localizaciones imprecisas y/o fallos de segmentación del sistema de visión artificial.
- Perturbaciones que produce la pinza cuando se encuentran fresas por debajo del objetivo o dentro del área de búsqueda de la pinza, por lo que la pinza detecta las fresas molestas y las considera objetivos. También debido a los toques que pueda tener el brazo robótico durante el proceso de recolección con las plantas, ya que estos toques afectan a la ubicación de objetivos.
- Región de alcance del robot reducida, ya que el espacio de trabajo con el que cuenta el brazo robótico para llevar a cabo la tarea de recolección es limitado.
- Fallos de comunicación del brazo o la pinza.

En 2020, la Universidad Politécnica Negeri Sriwijaya de Indonesia, analizó el empleo de un robot recolector como proyecto piloto, tal y como se cuenta el artículo [Oktarina et al., 2020], en el cual la fruta a recolectar serían tomates rojos y verdes, en lugar de fresas. A pesar de la diferencia en el tipo de cultivo, el estudio resulta relevante para este trabajo debido a la similitud en los principios de detección y recolección de frutos, y al tratarse de un prototipo básico lo convierte en un punto de partida útil para proyectos de investigación enfocados en la automatización agrícola aplicada a fresas. El objetivo de este estudio es establecer el proyecto inicial en la creación de una serie de robots aplicados en la agricultura para hacer realidad la idea de la agricultura digital. La novedad de este estudio es que este método es simple, al igual que el procesamiento de imágenes, para adaptarse a los recursos limitados de los procesadores con los que se llevó a cabo este proyecto. El robot diseñado se personaliza en función del tamaño de la tomatera y el color, donde la ubicación del tomate se indica mediante un círculo resultante del tratamiento de la imagen y la división en las posiciones izquierda, derecha y central en el plano de la imagen, cuya resolución en este estudio es de  $320 \times 180$  píxeles, como se muestra en la Figura 2.7.

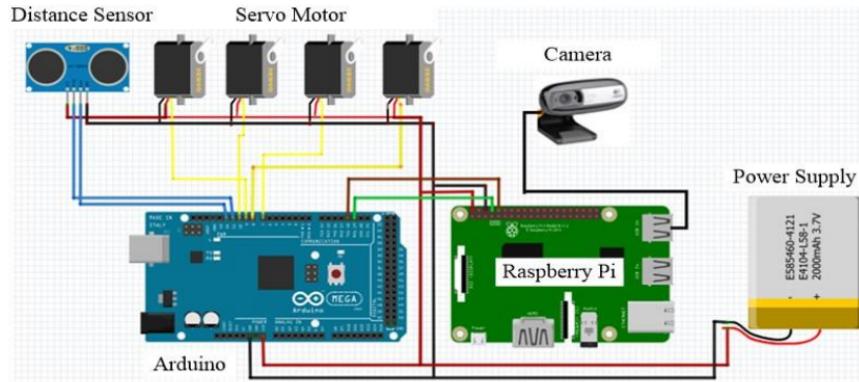


(a) Detección de tomates rojos

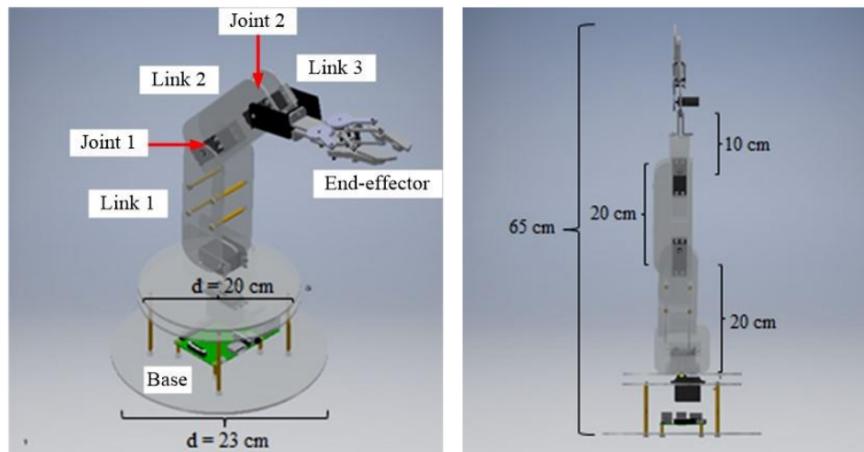
(b) Detección de tomates verdes

Figura 2.7: Detección de tomates indicada por los círculos rojos

El robot, representado en la Figura 2.8, consta de una cámara web común instalada en el efecto final y un sensor ultrasónico de proximidad HC-SR04 que puede detectar desde 2 cm hasta 400 cm, con un ángulo de detección de hasta  $30^\circ$  como entradas, siendo la cámara quién captura la imagen en bruto de la fruta, mientras que el sensor de proximidad detecta la distancia entre el efecto final, que consiste en una tijera para cortar la rama de tomates, y la fruta objetivo. El procesamiento de la imagen en bruto de la cámara se lleva a cabo en el procesador principal, que para este caso es un Arduino Mega 2560, y el resultado del procesamiento de imágenes (los tomates detectados) se envían a la Raspberry Pi 3 Modelo B, donde estos dan el número binario «1» correspondiente a los tomates detectados, mientras que al resto de la imagen, convertida en escala de grises, se considera «0», y se envían al microcontrolador los ángulos de los servomotores que mueven el robot para acercarse a los tomates. Este brazo robótico es un robot con 4 grados de libertad, formado por un servomotor para la base, dos articulaciones y un efecto final, donde los servomotores están conectados a las articulaciones y al efecto final, y el sensor de proximidad acoplado al efecto final proporciona la información al microcontrolador para garantizar la distancia de recogida correcta entre el efecto final y los tomates.



(a) Conexión eléctrica entre los componentes del robot recolector



(b) Diseño tridimensional del robot recolector

Figura 2.8: Robot recolector de tomates

Con todo esto, obtenemos un tiempo medio de recolección del tomate rojo de 4,932 s, y del tomate verde de 5,276 s, siendo el tiempo necesario para que el robot detecte el tomate rojo y vuelva a la posición de espera de 9,676 s, y de 10,586 s para los tomates verdes. Esta diferencia de tiempo se debe a la distancia del robot al tomate y no al color del mismo.

En este capítulo se han revisado algunos de los avances más relevantes en el campo de la robótica agrícola, más concretamente en los sistemas aplicados a la recolección de fresas, donde se ha evidenciado cómo el uso de la inteligencia artificial y la visión artificial ha impulsado mejoras significativas tanto en la velocidad como en la precisión de los procesos de recolección. Sin embargo, aún queda trabajo por hacer para optimizar y mejorar la eficiencia y la viabilidad económica de estos sistemas.

---

## **Capítulo 3**

# **Objetivos**

---

Una vez presentado el contexto general en el cual se enmarca el presente trabajo de fin de grado, en este capítulo se describen los objetivos y requisitos de este, así como la metodología y el plan de trabajo llevados a cabo.

### **3.1. Descripción del problema**

La necesidad de implementar soluciones tecnológicas que automaticen y optimicen las tareas de recolección incrementando la eficiencia en la recolección, mejorando la calidad del producto y disminuyendo los costes asociados, surge debido a la situación actual de la agricultura en la que, uno de los mayores desafíos que enfrenta es la recolección de frutas y hortalizas, problema que deriva de la escasa mano de obra disponible y el proceso manual que esto conlleva, y de la posibilidad de que existan errores humanos en la identificación de los frutos para su recolección, pudiendo influenciar esto en la calidad del producto, especialmente en la recolección de frutos que requieren un manejo cuidadoso como las fresas.

La solución propuesta en este trabajo busca ayudar a mejorar esta situación, proporcionando un robot de bajo coste y accesible para cualquier persona y que sirva para poder mejorar el proceso de reconocimiento por visión de la maduración de frutos, más concretamente fresas, para su posterior recolección. Por lo tanto, este proyecto pretende, como objetivo principal, utilizar un robot colaborativo que, gracias a su interfaz intuitiva y accesible para cualquier persona no acostumbrada a programar ni a la robótica, y junto con el sistema de detección elaborado con materiales de bajo coste, sea capaz de reconocer las fresas maduras de un sistema de cultivo agrícola vertical, para su posterior recolección por el brazo robótico gracias a la comunicación establecida entre el sistema de visión y el robot.

Con el fin de alcanzar este objetivo principal, se han establecido los siguientes subobjetivos:

1. Investigar las soluciones actuales que cumplen con las características y objetivos establecidos.
2. Seleccionar la técnica de inteligencia artificial de reconocimiento de frutas y seleccionar los componentes hardware necesarios para desarrollar el sistema de visión de bajo coste.
3. Optimizar la técnica escogida y adaptarla de tal manera que sea capaz de funcionar en nuestra plataforma. Al ser una técnica basada en Machine Learning, se deberá crear un dataset de valor con imágenes de fresas y, por lo tanto, hacer un correcto tratamiento de los datos para conseguir un resultado preciso en el posterior entrenamiento.
4. Realizar el entrenamiento con varios algoritmos de Machine Learning de clasificación. Estudiar el rendimiento y precisión de cada uno de ellos a través de pruebas con el sistema de visión y fresas reales.
5. Seleccionar el protocolo de comunicación entre el sistema de visión y el robot y llevar a cabo pruebas tanto simuladas a través del simulador que facilita el fabricante del robot como reales para establecer esta comunicación.
6. Programación tanto del robot como del archivo en python que posee el código del sistema de reconocimiento de las fresas y el guardado de las posiciones y la distancia de estas a la posición de la cámara para su posterior envío al brazo robótico.
7. Realización de pruebas de la aplicación final tanto en entornos simulados como reales.

## 3.2. Requisitos

Para solucionar los problemas descritos, además de cumplir los subobjetivos marcados, este trabajo deberá cumplir los siguientes requisitos:

1. Se utilizará *GNU/Linux*, con la distribución *Ubuntu 22.04 LTS* como sistema operativo en el hardware que se encargará de ejecutar el programa del sistema de visión.
2. Los modelos entrenados se deben ajustar a las limitaciones hardware que ejecutará el programa del sistema de visión.

3. El sistema deberá poder ser utilizado en tiempo real.
4. El *hardware* utilizado para el desarrollo del sistema de visión debe ser lo suficientemente económico para ser adquirido por cualquier estudiante.
5. La aplicación debe ser fácilmente reproducible y desplegable tanto en un entorno simulado como en un ambiente educativo real o de laboratorio.

### 3.3. Competencias

Las competencias empleadas del grado, que han sido tomadas de las distintas asignaturas del mismo, y que han sido utilizadas para la realización de este proyecto se dividen tanto en generales como específicas, y son las siguientes:

1. *Capacidad de organización y planificación: CG02.* Esta competencia ha sido empleada en la consecución de todo el trabajo de fin de grado, y queda reflejada tanto en las reuniones semanales o quincenales con el tutor responsable de este trabajo como en la wiki de GitHub<sup>13</sup> dedicada al proyecto, que refleja los avances y la organización llevada a cabo.
2. *Conocimiento de una lengua extranjera: CG04.* Esta competencia ha sido utilizada a la hora de buscar toda clase de información para poder elaborar este proyecto, ya que se han utilizado documentos publicados en, al menos, una lengua extranjera, como lo es el inglés.
3. *Resolución de problemas: CG06.* Dado el nivel de conocimiento necesario en ciertas materias como la inteligencia artificial o la visión artificial, ha sido empleada para poder resolver los diversos inconvenientes que afrontar las pruebas prácticas relacionadas con estas áreas del trabajo suponían.
4. *Uso de internet como medio de comunicación y como fuente de información: CG21.* Para poder elaborar este trabajo de fin de grado, ha sido necesario emplear esta competencia para buscar la información necesaria y poder completar principalmente los capítulos 1 y 2.
5. *Conocimientos de informática relativos al ámbito de estudio: CG24.* Esta competencia ha sido empleada a la hora de desarrollar y programar la aplicación sobre la que trata este trabajo.

---

<sup>13</sup><https://github.com/RoboticsURJC/tfg-dcampoamor/wiki>

6. *Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería: CE3.* Para poder desarrollar la aplicación se tuvo que emplear esta competencia a la hora de llevar a cabo la partición del disco duro en el ordenador y poder instalar la versión de *Ubuntu 22.04.5 LTS (Jammy Jellyfish)*.
7. *Conocimientos sobre los fundamentos de automatismos y métodos de control: CE13.* Esta competencia se refleja en la programación de la toma de decisiones automática, donde el sistema ajusta las operaciones en función de los resultados obtenidos del sistema de visión, así como el trabajo sincronizado de varios dispositivos (cámara y robot) y la comunicación entre estos.
8. *Conocimiento de los principios de regulación automática y su aplicación a la automatización industrial: CE32.* Esta competencia se emplea una vez que el sistema de visión identifica el grado de maduración de la fresa, ya que el sistema toma la decisión de recolectar o no el fruto, ajustando los actuadores o robots para realizar la tarea de forma precisa, dependiendo de las condiciones detectadas. Así mismo, se emplea la regulación automática en la optimización del proceso ajustando el comportamiento del robot en función del estado de la maduración de la fresa, maximizando la velocidad de la aplicación y su precisión.
9. *Capacidad para diseñar sistemas de control y automatización industrial: CE33.* Gracias a esta competencia se ha podido estructurar el sistema completo, incluyendo la parte de visión artificial, el procesamiento de imágenes, la toma de decisiones y el control del brazo robótico, integrando todos los datos en tiempo real y llevando a cabo operaciones de monitoreo y seguimiento de la aplicación y sus operaciones en remoto, tal y como se realizó en la etapa de pruebas.

Por otro lado, las competencias adquiridas con el desarrollo de este trabajo fin de grado, y que aparecen descritas en la guía docente de la propia asignatura, son las siguientes:

1. *Capacidad de análisis y síntesis: CG01.* Esta competencia se adquiere debido a la necesidad de la búsqueda y recopilación de información necesaria para el desarrollo del proyecto.
2. *Razonamiento crítico: CG11.* En relación con la competencia adquirida CG01, el razonamiento crítico se emplea a la hora de filtrar, seleccionar y decidir qué de

toda la información obtenida es válido, cómo se puede utilizar y cómo ajustarlo y añadirlo al contenido del proyecto.

3. *Aprendizaje autónomo: CG13.* Esta competencia ha sido adquirida dada la necesidad de adaptarse al marco técnico en el cual se desarrolla este proyecto, su complejidad, y la constante actualización y mejoras de las técnicas que pueden ser empleadas para el desarrollo del mismo.
4. *Adaptación a nuevas situaciones: CG14.* Esta competencia se adquiere gracias a la aplicación de la competencia adquirida *CG13 Aprendizaje autónomo*, justificada anteriormente, y que se puede ver reflejada en el proyecto.
5. *Capacidad de aplicar los conocimientos teóricos en la práctica: CG20.* La parte empírica y práctica del proyecto refleja la adquisición de esta competencia.
6. *Capacidad para entender el lenguaje y propuestas de otros especialistas: CG22.* Esta competencia se ha visto reflejada dado que, en relación con el resto de competencias adquiridas, sobre todo con las competencias *CG01 Capacidad de análisis y síntesis* y *CG11 Razonamiento crítico*, se ha tenido la necesidad de consulta, recopilación y filtrado de información y metodologías científicas existentes para poder desarrollar la aplicación del proyecto.

### 3.4. Metodología

Para lograr los objetivos descritos anteriormente, se optó por emplear el método DMADV (Definir, Medir, Analizar, Diseñar, Verificar), perteneciente a la metodología *Six Sigma* (ver en Figura 3.1).

Se decidió optar por esta metodología dado que la aplicación desarrollada en este trabajo está basada en otros proyectos y estudios similares ya existentes, y esta metodología está diseñada para desarrollar procesos o productos nuevos o significativamente mejorados, independientemente de si partes de cero o si te basas en conocimientos previos, y esta información recopilada fue utilizada como punto de partida en las fases iniciales de la metodología. A continuación, se detallan las fases del método DMADV y cómo han sido aplicadas en el desarrollo de este proyecto.

- Definir: En esta fase se establecen como objetivos del proyecto la identificación de fresas maduras mediante un sistema de visión artificial y su integración con el brazo robótico encargado de su recolección.

- Medir: Tomando como referencia el estado del arte existente, se llevó a cabo la selección de la tecnología (hardware y software necesarios) y las métricas existentes para poder considerar que las detecciones son aceptables y suficientes, basándose así el trabajo en proyectos y estudios que han demostrado ser eficaces, constituyendo un proyecto robusto y eficiente y ayudando a evitar problemas comunes ya que se sostiene sobre conocimientos adquiridos de proyectos similares testados.
- Analizar: En esta etapa se llevaron a cabo pruebas con diferentes algoritmos y sistemas para detección de objetos, tanto en imágenes como en vídeo a tiempo real, para poder determinar cuál ofrecía un mejor rendimiento en precisión y velocidad y se ajustaba a los requisitos que presentaba el hardware disponible para el desarrollo del proyecto.
- Diseñar: En base a los resultados obtenidos en la fase de análisis, se configuró el sistema de reconocimiento por visión incluyendo el hardware, previamente seleccionado, y el software, compuesto por los códigos y entornos necesarios para poder desarrollarlo y ejecutarlo, y su integración con el robot.
- Verificar: Finalmente, se llevaron a cabo pruebas para verificar que el sistema funcionaba y cumplía con los requisitos definidos, validándose la precisión en la identificación de las fresas maduras con distintas intensidades de luz ambiente y la comunicación y el correcto funcionamiento del brazo robótico.

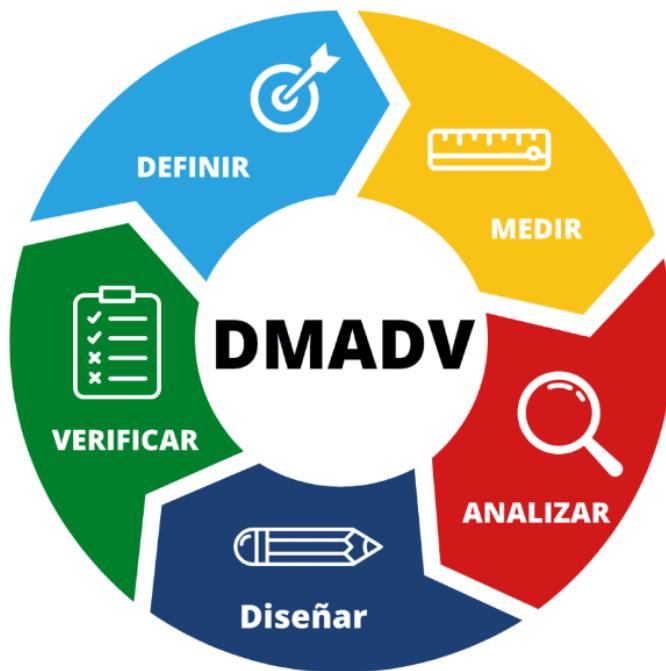


Figura 3.1: Ciclo de la metodología DMADV

### 3.5. Plan de trabajo

El desarrollo y seguimiento que el proyecto ha seguido es una planificación en base a reuniones semanales con el tutor, en las cuales se revisaron los avances, se fijaron nuevos objetivos y se discutieron y propusieron posibles mejoras, mientras que el trabajo se organizó en varias fases clave:

1. *Investigación inicial:* En esta fase, se investigó el estado del arte relacionado con sistemas de visión artificial y técnicas de reconocimiento de objetos, especialmente aplicadas a la maduración de frutas y hortalizas, y utilizando para ello artículos científicos, capítulos de libros y proyectos previos.
2. *Diseño y desarrollo del sistema de visión artificial:* Esta fase se centró en el diseño y la implementación del sistema de visión artificial, abarcando tanto el desarrollo del software como la integración del hardware, e incluyendo la calibración y obtención de los parámetros intrínsecos a la cámara y las diversas pruebas realizadas con distintos sistemas y códigos, hasta seleccionar el *software* funcional con el que se llevó a cabo el proyecto finalmente.
3. *Pruebas en entorno simulado:* Durante esta fase se realizaron múltiples pruebas y ajustes para optimizar el funcionamiento del sistema y comprobar su funcionamiento en diferentes escenarios, simulando de manera separada la programación del robot, para el que se utilizó un simulador en una máquina virtual, y la detección y funcionamiento del sistema de visión, cuyos algoritmos se afinaron para mejorar la precisión en la detección y se ajustaron los parámetros relacionados con la cámara en los códigos para poder obtener las coordenadas y distancia real de las detecciones respecto a la cámara y poder transmitírselas al brazo robótico. Finalmente, también se llevaron a cabo pruebas de comunicación entre el sistema de visión y el robot, poniendo a prueba su programación, para que este alcanzase el punto de la detección.
4. *Pruebas en entorno real:* Una vez desarrollado el prototipo inicial, el sistema completo fue sometido a pruebas en un entorno real de lo que sería la aplicación final.
5. *Escritura de la memoria:* Con el sistema ya afinado y probado, se procedió a la redacción de la memoria del proyecto. En esta etapa, se documentó detalladamente todo el proceso seguido, desde la investigación inicial hasta los resultados finales obtenidos durante las pruebas reales.

Todo el contenido del proyecto se puede encontrar en un repositorio público de GitHub<sup>14</sup>, donde en el apartado Wiki<sup>15</sup> de dicho repositorio puede verse el desarrollo del trabajo en semanas a lo largo de los meses, durante el tránscurso del proyecto.

Después de haber revisado los objetivos, requisitos, competencias, metodología y el plan de trabajo implementado para la realización de este proyecto, se abordarán las plataformas de desarrollo empleadas.

---

<sup>14</sup><https://github.com/RoboticsURJC/tfg-dcampoamor>

<sup>15</sup><https://github.com/RoboticsURJC/tfg-dcampoamor/wiki>

---

## Capítulo 4

# Plataforma de desarrollo

---

Con los objetivos del proyecto definidos, en este capítulo se abordarán las distintas plataformas de desarrollo, tanto *hardware* como *software*, que han facilitado el logro de esos objetivos.

### 4.1. Hardware

Este apartado recoge la descripción de los componentes *hardware* utilizados en este proyecto, para los cuales se ha buscado priorizar la reducción de costes en cada elección y utilizar aquellos elementos a los que se tenía acceso a la hora del desarrollo del proyecto o de los que se disponía a la hora de elaborarlo.

#### 4.1.1. Cámara Logitech C270 HD

Esta cámara web (Figura 4.1) de dimensiones 72,91 x 31,91 x 66,64 mm, corrige la iluminación de manera automática, produciendo colores reales y naturales y ajustándose a las condiciones de iluminación del entorno, lo que facilita la detección de fresas. Ofrece una resolución HD 720p, proporcionando imágenes claras y nítidas a una velocidad de 30 fotogramas por segundo (fps) con una lente que cuenta con enfoque fijo y un campo visual diagonal (dFoV) de 55 grados. Su coste aproximado es, en la página oficial, de 44€, aunque en otras empresas destacadas en el sector de la electrónica y la tecnología en España puede encontrarse por, aproximadamente, 25€.



Figura 4.1: Cámara Logitech C270 HD<sup>16</sup>

#### 4.1.2. Soporte de brazo articulado

Para poder ubicar la cámara en una posición fija que permitiese visualizar las fresas, se utilizó un soporte de brazo articulado, como el representado en la Figura 4.2, cuya parte fija en la parte inferior se ancla a la mesa. Este soporte articulado, tiene un ajuste de 360 grados, con su extremo más largo de 75 cm, mientras que la carga máxima que permite es de 560 gramos cuando se coloca de manera horizontal, siendo el precio de este soporte 22,98€.



Figura 4.2: Soporte de brazo articulado<sup>17</sup>

#### 4.1.3. Ordenador principal

El equipo que se ha configurado como entorno de trabajo para este proyecto es el que aparece en la Figura 4.3, sirviendo como base para el desarrollo de la programación y las pruebas de la visión artificial, y utilizándose igualmente como servidor para poder llevar a cabo la comunicación con el brazo robótico mediante el protocolo XML-RPC basado en HTTP, y posteriormente el envío de posiciones detectadas en tiempo real a este.

<sup>16</sup><https://www.logitech.com/es-es/products/webcams/c270-hd-webcam.960-001063.html?srsltid=AfmB0or4HptUTcGrxE-4SzxKR-ARw-ykNeagHSezXUvT1Xkx8qLfY41G>

<sup>17</sup>[https://www.amazon.es/dp/B08JCG4V5S?ref=ppx\\_pop\\_mob\\_ap\\_share&th=1](https://www.amazon.es/dp/B08JCG4V5S?ref=ppx_pop_mob_ap_share&th=1)

Figura 4.3: Lenovo Legion 5 15IMH05<sup>18</sup>

A continuación, en el Cuadro 4.1, se recogen las características técnicas del ordenador utilizado:

Características	Descripción
Pantalla	15,6 pulgadas Full HD (1920x1080)
Procesador (CPU)	Intel Core i7-10750H CPU @ 2.60GHz
Memoria RAM	16 GB
Almacenamiento	1 TB
Tarjeta gráfica (GPU)	NVIDIA GeForce GTX 1650 Ti Mobile
Sistema Operativo	Windows 10 y Ubuntu 22.04.4 LTS
Cámara de portátil	HD 720p con tapa de privacidad
Puertos	4 x USB 3.2 Gen 1 Tipo-A 1 x USB 3.2 Gen 1 Tipo-C 1 x Ethernet (RJ-45) 1 x HDMI 2.0 1 x combo auriculares/micrófono (3.5 mm)
Conectividad	WiFi 6 802.11ax (2x2), Bluetooth 5.0, Ethernet 100/1000M
Batería	80 Wh
Peso	2,3 kg
Dimensiones	363,06 x 259,61 x 23,57–26,1 mm

Cuadro 4.1: Especificaciones técnicas del ordenador usado

---

<sup>18</sup><https://www.pcccomponentes.com/lenovo-legion-5-15imh05-intel-core-i7-10750h-16gb-1tb-ssd-gtx-1650-156?srsltid=AfmBOopJpvGSHUyQU696jgG7-6orSKMOEWZe2ZvvYtA0NGtJ9Ms2xJFp>

#### 4.1.4. Robot de Universal Robots de la gama e-series

Los robots de Universal Robots, también conocidos como robots colaborativos o cobots, están diseñados para trabajar junto a los humanos de manera segura, eficiente y flexible tanto en aplicaciones industriales como no industriales, destacando por su facilidad de uso, versatilidad y capacidad para automatizar tareas repetitivas o peligrosas.

Fabricados en aluminio junto con otros materiales de bajo peso, permitiendo reducir la inercia del propio robot y facilitar su manipulación, cada brazo robótico tiene seis ejes, otorgando al robot seis grados de libertad (Degree Of Freedom o DOF), que permiten movimientos precisos y fluidos, y en cuyas articulaciones están equipadas, a su vez, encoders absolutos, reductoras armónicas, que reducen la velocidad de rotación de los engranajes en las juntas, y aumentan el par del eje, ofreciendo una alta precisión y eficiencia; y, en aquellos brazos robóticos pertenecientes a la gama e-series, sensores de fuerza y torque, encontrándose integrados en el efecto o tool flange del propio brazo robótico <sup>19</sup>.

Por otro lado, en la controladora de los brazos robóticos de la gama e-series, pueden diferenciarse varios módulos: la unidad de procesamiento principal (CPU), que se encarga de realizar los cálculos cinemáticos y dinámicos del robot, la ejecución de programas y control del brazo robótico mediante su software Polyscope, y la comunicación con los periféricos y la consola de mando o teach pendant; la fuente de alimentación, que suministra energía a todos los componentes del sistema, incluyendo a las articulaciones del brazo robótico y el teach pendant, y que incluye protección contra sobrecargas y cortocircuitos para garantizar una alimentación estable y segura; el módulo de seguridad o la placa de seguridad (Safety Control Board o SCB), que implementa funciones de seguridad colaborativa, como límites de velocidad, fuerza y espacio de trabajo, y que supervisa las entradas de seguridad y gestiona las paradas de emergencia; las interfaces de comunicación, compuesta del puerto ethernet, que soporta los protocolos de comunicación ModbusTCP, ProfiNet y EthernetIP, los dos puertos USB (uno de ellos USB 2.0 y otro USB 3.0), las dieciseis entradas y salidas digitales, y las dos entradas y salidas analógicas; el sistema de refrigeración, que está compuesto por ventiladores internos, que están controlados por sensores térmicos que ajustan su velocidad en función de la temperatura interna de los componentes, llegando a activar alarmas si se detectasen temperaturas anormales, rejillas de ventilación en el lateral de la carcasa o envolvente de acero que recubre y protege los componentes internos de la controladora, que a su

<sup>19</sup><https://www.universal-robots.com/mx/acerca-de-universal-robots/noticias/meet-the-next-generation-of-collaborative-ur-robots-at-robobusiness>

vez están protegidas por filtros para evitar la entrada de polvo o partículas grandes; y por último, del módulo de almacenamiento interno, que para esta gama de robots se compone de una tarjeta SSD.

Para el desarrollo de este proyecto se han utilizado diferentes modelos de robots de la gama e-series, todos ellos representados en la Figura 4.4, desde el UR3e hasta el UR10e, pasando por el UR5e. A pesar de que tanto la gama CB-series como la gama e-series permiten la comunicación mediante el protocolo XML-RPC, la interfaz más intuitiva y simplificada que facilita la programación y reduce la sobrecarga de información, poseer sensor de fuerza integrado, permitiendo un control más preciso y sensible haciendo que mejore la interacción con el entorno, así como una mayor precisión en la repetición de movimientos, entre otros factores, constituyeron que se eligiera la gama e-series para este trabajo.

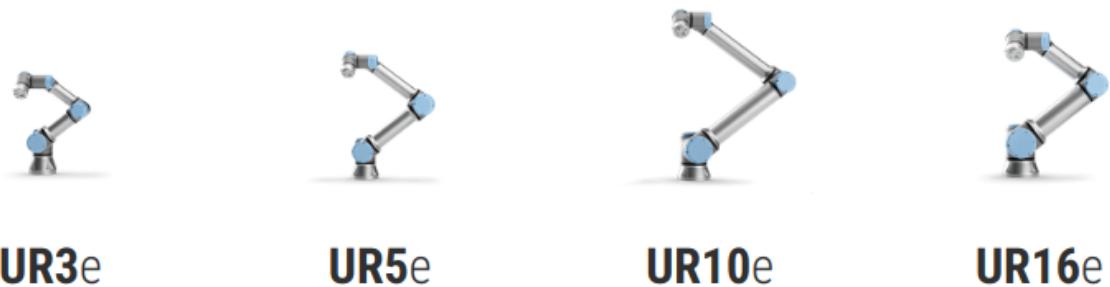


Figura 4.4: Gama e-series de Universal Robots<sup>19</sup>

## 4.2. Software

Este apartado estará dedicado a detallar las plataformas de software, librerías y entornos de trabajo que han sido fundamentales para alcanzar los objetivos definidos en el Capítulo 3, desde el sistema operativo utilizado hasta las tecnologías específicas para el procesamiento de imágenes y aprendizaje profundo.

### 4.2.1. Ubuntu

Ubuntu<sup>20</sup> es un sistema operativo de código abierto basado en Linux y desarrollado por la empresa británica Canonical Ltd. Este sistema operativo está diseñado para ser utilizado en una gran variedad de dispositivos, y es reconocido por su facilidad de

<sup>19</sup><https://www.universal-robots.com/es/productos/>

<sup>20</sup><https://ubuntu.com/>

uso, estabilidad y seguridad, contando con una amplia comunidad de desarrolladores y usuarios que contribuyen activamente a su desarrollo y soporte. La versión utilizada para la realización de este proyecto, de entre todas las versiones disponibles, tal y como muestra Figura 4.5, es Ubuntu 22.04 Long Term Support (LTS) (Jammy Jellyfish), ya que era la última versión disponible de Ubuntu en el momento en el que se empezó a elaborar el proyecto. Sus principales diferencias con versiones anteriores incluyen el uso de un kernel y versiones de python más modernas, que proporcionan mejor soporte para hardware reciente y optimizaciones de rendimiento, un entorno de escritorio GNOME, que ha evolucionado de Unity en versiones antiguas como 14.04 y 16.04, y la reducción en el tamaño de las imágenes del sistema, al igual que la optimización del uso de recursos.

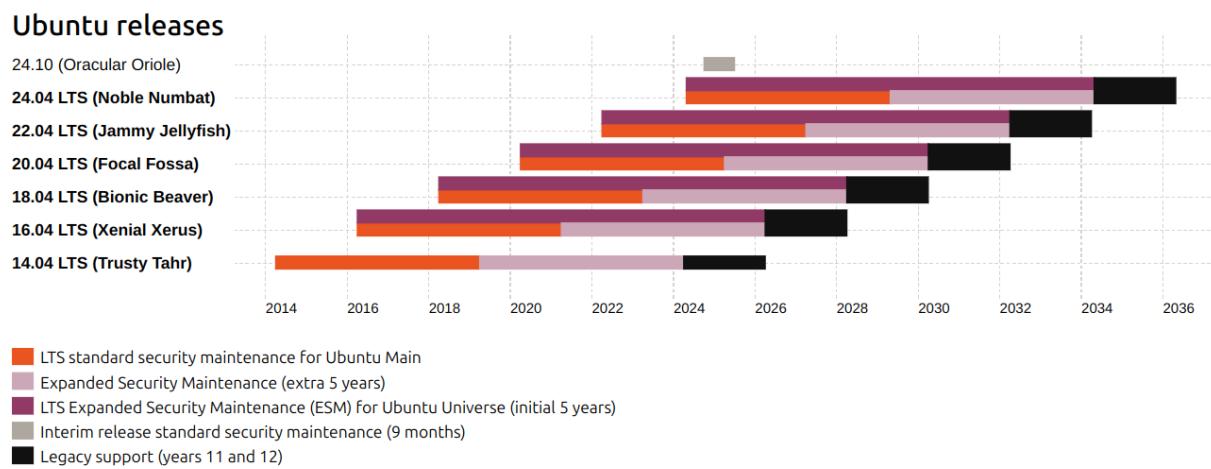


Figura 4.5: Lanzamiento de las versiones de Ubuntu

### 4.2.2. Polyscope

### 4.2.3. Python

Python<sup>21</sup>(Figura 4.6) es un lenguaje de programación de alto nivel, orientado a objetos y de semántica dinámica. La sintaxis de Python, sencilla y fácil de aprender, favorece la legibilidad y, por tanto, reduce el coste de mantenimiento de los programas, admitiendo módulos y paquetes, lo que fomenta la modularidad del programa y la reutilización de código <sup>22</sup> para el desarrollo de aplicaciones en diferentes áreas, como sucede en este caso con la inteligencia y visión artificial y el aprendizaje automático (PyTorch, NumPy y OpenCV).



Figura 4.6: Logo de Python

### 4.2.4. PyTorch

De los desarrolladores de Facebook AI Research junto a otros laboratorios, PyTorch<sup>23</sup> es un marco de aprendizaje profundo de código abierto conocido por su compatibilidad con Python, siendo un marco de trabajo completo para crear modelos de aprendizaje profundo. Se distingue por su excelente compatibilidad con GPU y su uso de la autodiferenciación en modo inverso, que permite modificar los gráficos de cálculo sobre la marcha, lo que lo convierte en una opción popular para la experimentación rápida y la creación de prototipos.

### 4.2.5. NumPy

NumPy (Numerical Python)<sup>24</sup> es una biblioteca de Python fundamental para el cálculo numérico, que proporciona un objeto de matriz multidimensional, varios objetos derivados (como matrices y matrices enmascaradas) y un surtido de rutinas para realizar operaciones rápidas con matrices <sup>25</sup>.

<sup>21</sup><https://www.python.org/>

<sup>22</sup><https://www.python.org/doc/essays/blurb/>

<sup>23</sup><https://pytorch.org/>

<sup>24</sup><https://numpy.org/>

<sup>25</sup><https://numpy.org/doc/stable/user/whatisnumpy.html>

En este proyecto, la biblioteca Numpy se ha utilizado para inicializar matrices y vectores de los parámetros intrínsecos y extrínsecos de la cámara, realizar cálculos geométricos y poder obtener las matrices de rotación y traslación de la cámara, llevar a cabo operaciones matriciales como multiplicaciones e inversiones, y para poder proyectar las coordenadas en el espacio 2D a coordenadas tridimensionales mediante operaciones matriciales, pudiendo obtener posteriormente las distancias a los objetos detectados.

#### 4.2.6. OpenCV

OpenCV (Open Source Computer Vision Library)<sup>26</sup> es una biblioteca de software de código abierto diseñada para su uso en aplicaciones de aprendizaje automático y visión artificial. Desarrollado por Intel<sup>27</sup> en 1999, cuenta con más de 2.500 algoritmos optimizados, que incluyen un amplio conjunto de algoritmos de visión por ordenador y aprendizaje automático, que permiten desde el reconocimiento y detección de caras, objetos y acciones humanas en vídeos, hasta el seguimiento de movimientos de estos a tiempo real en vídeo. Además, incluyen herramientas para la creación de modelos y nubes de puntos 3D, la generación de imágenes de alta resolución a partir de múltiples tomas, y la búsqueda de imágenes similares en bases de datos. También ofrecen funcionalidades como la eliminación de ojos rojos, el seguimiento ocular, el reconocimiento de paisajes, y la superposición de realidad aumentada mediante marcadores.

En este proyecto, OpenCV (importado como cv2) se ha usado para poder capturar los frames desde la cámara en tiempo real y mostrarlo en una ventana para poder controlar y verificar que se realizan las detecciones correctamente, para convertir estos frames del formato BGR a RGB, dibujar un rectángulo alrededor de las fresas detectadas, añadiendo la etiqueta de texto correspondiente que indica la clase detectada y la confianza de esta detección; permite detener el bucle principal del programa, terminarlo y salir de este si se presiona la tecla configurada para ello, asegurando que la cámara o el archivo de vídeo no permanezcan bloqueados por el programa y cerrando todas las ventanas de visualización creadas.

---

<sup>26</sup><https://opencv.org/>

<sup>27</sup><https://www.intel.es>

#### 4.2.7. XML-RPC

XML-RPC<sup>28</sup> es un método de Llamada a Procedimiento Remoto (o RPC en inglés) que usa XML para codificar y transferir datos entre programas a través de sockets y HTTP como protocolo de transporte. Para muchos lenguajes de programación existen servidores XML-RPC gratuitos, entre otros para: Python, Java, C++ y C.

Debido al uso de Python en el proyecto, se utilizó el paquete *xmlrpc*, que agrupa los módulos tanto de cliente como de servidor que implementan XML-RPC. Con este paquete, el controlador del robot UR puede llamar a métodos/funciones (con parámetros) en un programa/servidor remoto y obtener de vuelta datos estructurados, pudiendo realizar un cálculo complejo mediante su uso, que no está disponible en el lenguaje propio de programación del robot.

#### 4.2.8. Anaconda

Anaconda<sup>29</sup> es una distribución de código abierto para los lenguajes de programación Python y R, diseñada para facilitar la gestión de paquetes y entornos, así como el despliegue de aplicaciones de ciencia de datos y aprendizaje automático. Ofrece herramientas como conda, un sistema de gestión de paquetes y entornos que funciona en Windows, macOS y Linux; y Anaconda Navigator, una aplicación de escritorio que permite gestionar aplicaciones integradas, paquetes y entornos sin necesidad de utilizar la línea de comandos <sup>30</sup>.

Para este proyecto se ha hecho uso del programa conda, ya que, utilizando esta herramienta es posible instalar y actualizar paquetes y dependencias y cambiar entre entornos desde el mismo ordenador local, permitiendo que puedan ser mantenidos y ejecutados independientemente sin archivos, directorios y rutas, para que se pueda trabajar con versiones específicas de librerías y/o Python mismo, sin afectar a otros proyectos Python, es decir, no afectando los cambios de un entorno a otros <sup>31</sup>.

---

<sup>28</sup><https://docs.python.org/es/3.8/library/xmlrpc.html>

<sup>29</sup><https://www.anaconda.com/>

<sup>30</sup><https://docs.anaconda.com/anaconda/>

<sup>31</sup><https://docs.anaconda.com/reference/glossary/#conda>

#### 4.2.9. YOLOv3

YOLOv3 (You Only Look Once versión 3)<sup>32</sup> es un algoritmo de detección de objetos en tiempo real que identifica y localiza múltiples objetos dentro de una imagen o video. Desarrollado por Joseph Redmon y Ali Farhadi en 2018, YOLOv3 es la tercera iteración de la serie YOLO (Figura 4.7), conocida por su capacidad para realizar detecciones rápidas y precisas [Redmon and Farhadi, 2018]. Introdujo el uso de tres escalas diferentes para la detección, aprovechando tres tamaños distintos de núcleos de detección: 13x13, 26x26 y 52x52, lo que mejoró significativamente la precisión de la detección de objetos de diferentes tamaños <sup>33</sup>. La serie YOLOv3, está diseñada específicamente para tareas de detección de objetos, además, YOLOv3 añadió funciones como predicciones multietiqueta para cada cuadro delimitador y una red extractora de características mejorada. Estos modelos son famosos por su eficacia en diversos escenarios del mundo real, equilibrando precisión y velocidad, lo que los hace adecuados para una amplia gama de aplicaciones <sup>34</sup>.

Esta herramienta ha sido elegida para el entrenamiento del modelo de detección de fresas debido a su eficiencia en el uso de recursos computacionales, lo que lo hace más accesible para sistemas con capacidades limitadas; a la amplia documentación y la comunidad activa existente en torno a YOLOv3, que proporciona recursos valiosos para la implementación y resolución de problemas, lo que es esencial en proyectos académicos con plazos definidos <sup>35</sup>; a la competencia en cuanto a precisión y velocidad de YOLOv3 frente a versiones superiores a pesar de presentar estas ciertas mejoras; y a que YOLOv3 es compatible con entornos de desarrollo ampliamente utilizados en proyectos académicos, como Python y bibliotecas estándar de aprendizaje automático, facilitando su integración en el flujo de trabajo del proyecto.

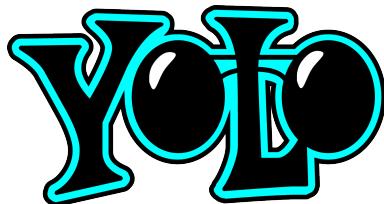


Figura 4.7: Logo de YOLO

<sup>32</sup><https://docs.ultralytics.com/es/models/yolov3/>

<sup>33</sup><https://docs.ultralytics.com/es/models/yolov3/#key-features>

<sup>34</sup><https://docs.ultralytics.com/es/models/yolov3/#supported-tasks-and-modes>

<sup>35</sup><https://github.com/ultralytics/yolov3/>

Una vez analizadas las plataformas de software y hardware utilizadas en este trabajo de fin de grado, se procederá a detallar el proceso completo de diseño y desarrollo de la aplicación, lo cual será explicado con detalle en los capítulos siguientes.

---

# Capítulo 5

## Diseño

---

*Quizás algún fragmento de libro inspirador...*

Autor, Título

Escribe aquí un párrafo explicando brevemente lo que vas a contar en este capítulo. En este capítulo (y quizás alguno más) es donde, por fin, describes detalladamente qué has hecho y qué experimentos has llevado a cabo para validar tus desarrollos.

### 5.1. Snippets

Puede resultar interesante, para clarificar la descripción, mostrar fragmentos de código (o *snippets*) ilustrativos. En el Código 5.1 vemos un ejemplo escrito en C++.

---

```
void Memory::hypothesizeParallelograms () {
    for(it1 = this->controller->segmentMemory.begin(); it1++) {
        squareFound = false; it2 = it1; it2++;
        while ((it2 != this->controller->segmentMemory.end()) && (!squareFound))
            {
                if (geometry::haveACommonVertex((*it1), (*it2), &square)) {
                    dist1 = geometry::distanceBetweenPoints3D ((*it1).start, (*it1).end);
                    dist2 = geometry::distanceBetweenPoints3D ((*it2).start, (*it2).end);
                }
            // [...]
```

---

Código 5.1: Función para buscar elementos 3D en la imagen

En el Código 5.2 vemos un ejemplo escrito en Python.

### 5.2. Verbatim

Para mencionar identificadores usados en el código —como nombres de funciones o variables— en el texto, usa el entorno literal o verbatim `hypothesizeParallelograms()`. También se puede usar este entorno para varias líneas, como se ve a continuación:

---

```

def mostrarValores():
    print (w1.get(), w2.get())

master = Tk()
w1 = Scale(master, from_=0, to=42)
w1.pack()
w2 = Scale(master, from_=0, to=200, orient=HORIZONTAL)
w2.pack()
Button(master, text='Show', command=mostrarValores).pack()

mainloop()

```

---

Código 5.2: Cómo usar un Slider

```

void Memory::hypothesizeParallelograms () {
    // add your code here
}

```

### 5.3. Ecuaciones

Si necesitas insertar alguna ecuación, puedes hacerlo. Al igual que las figuras, no te olvides de referenciarlas. A continuación se exponen algunas ecuaciones de ejemplo: Ecuación 5.1 y Ecuación 5.2.

$$H = 1 - \frac{\sum_{i=0}^N \frac{(\frac{d_{js} + d_{je}}{2})}{N}}{M} \quad (5.1)$$

Ecuación 5.1: Ejemplo de ecuación con fracciones

$$v(\text{entrada}) = \begin{cases} 0 & \text{if } \epsilon_t < 0,1 \\ K_p \cdot (T_t - T) & \text{if } 0,1 \leq \epsilon_t < M_t \\ K_p \cdot M_t & \text{if } M_t < \epsilon_t \end{cases} \quad (5.2)$$

Ecuación 5.2: Ejemplo de ecuación con array y letras y símbolos especiales

### 5.4. Tablas o cuadros

Si necesitas insertar una tabla, hazlo dignamente usando las propias tablas de L<sup>A</sup>T<sub>E</sub>X, no usando pantallazos e insertándolas como figuras... En el Cuadro 5.1 vemos un ejemplo.

Parámetros	Valores
Tipo de sensor	Sony IMX219PQ[7] CMOS 8-Mpx
Tamaño del sensor	3.674 x 2.760 mm (1/4"format)
Número de pixels	3280 x 2464 (active pixels)
Tamaño de pixel	1.12 x 1.12 um
Lente	f=3.04 mm, f/2.0
Ángulo de visión	62.2 x 48.8 degrees
Lente SLR equivalente	29 mm

Cuadro 5.1: Parámetros intrínsecos de la cámara

---

# Capítulo 6

# Conclusiones

---

*Quizás algún fragmento de libro inspirador...*

Autor, Título

Escribe aquí un párrafo explicando brevemente lo que vas a contar en este capítulo, que básicamente será una recapitulación de los problemas que has abordado, las soluciones que has prouesto, así como los experimentos llevados a cabo para validarlos. Y con esto, cierras la memoria.

## 6.1. Conclusiones

Enumera los objetivos y cómo los has cumplido.

Enumera también los requisitos implícitos en la consecución de esos objetivos, y cómo se han satisfecho.

No olvides dedicar un par de párrafos para hacer un balance global de qué has conseguido, y por qué es un avance respecto a lo que tenías inicialmente. Haz mención expresa de alguna limitación o peculiaridad de tu sistema y por qué es así. Y también, qué has aprendido desarrollando este trabajo.

Por último, añade otro par de párrafos de líneas futuras; esto es, cómo se puede continuar tu trabajo para abarcar una solución más amplia, o qué otras ramas de la investigación podrían seguirse partiendo de este trabajo, o cómo se podría mejorar para conseguir una aplicación real de este desarrollo (si es que no se ha llegado a conseguir).

## 6.2. Corrector ortográfico

Una vez tengas todo, no olvides pasar el corrector ortográfico de L<sup>A</sup>T<sub>E</sub>Xa todos tus ficheros *.tex*. En Windows, el propio editor TeXworks incluye el corrector. En Linux, usa aspell ejecutando el siguiente comando en tu terminal:

```
aspell --lang=es --mode=tex check capitulo1.tex
```

# Bibliografía

---

- [Abeliuk and Gutiérrez, 2021] Abeliuk, A. and Gutiérrez, C. (2021). Historia y evolución de la inteligencia artificial. *Revista Bits de Ciencia*, (21):14–21.
- [Aizenberg et al., 2000] Aizenberg, I., Aizenberg, N. N., and Vandewalle, J. P. (2000). *Multi-valued and universal binary neurons: Theory, learning and applications*. Springer Science & Business Media. First work to introduce the term Deep Learning to Neural Networks.
- [Alvear-Puertas et al., 2017] Alvear-Puertas, V., Rosero-Montalvo, P., Peluffo-Ordóñez, D., and Pijal-Rojas, J. (2017). Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura. *Enfoque UTE*, 8:244–256.
- [Aracil et al., 2008] Aracil, R., Balaguer, C., and Armada, M. (2008). Robots de servicio. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 5(2):6–13.
- [Arnold et al., 2011] Arnold, L., Rebecchi, S., Chevalier, S., and Paugam-Moisy, H. (2011). An Introduction to Deep Learning. In *European Symposium on Artificial Neural Networks (ESANN)*, Proceedings of the European Symposium on Artificial Neural Networks (ESANN), Bruges, Belgium. HAL open science.
- [Barrientos, 2002] Barrientos, A. (2002). Nuevas aplicaciones de la robótica. robots de servicio. *Avances en robótica y visión por computador*. Cuenca, Ediciones Castilla-La Mancha, 288.
- [Basogain, 2008] Basogain, X. (2008). Redes neuronales artificiales y sus aplicaciones. *Dpto. Ingeniería de Sistemas y Automática, Escuela Superior de Ingeniería Bilbao.*, page 79. Open Course Ware. [En línea] dispnible en <http://ocw.ehu.es/ensenanzas-tecnicas/redes-neuronales-artificiales-y-sus-aplicaciones/Course-listing>.
- [Beasley, 2012] Beasley, R. A. (2012). Medical Robots: Current Systems and Research Directions. *Journal of Robotics*.

- [Bischoff et al., 2010] Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schäffer, A., Beyer, A., Eiberger, O., Haddadin, S., Stemmer, A., Grunwald, G., and Hirzinger, G. (2010). The KUKA-DLR Lightweight Robot arm - A new reference platform for robotics research and manufacturing. volume 2, pages 1–8.
- [Bradski and Kaehler, 2008] Bradski, G. and Kaehler, A. (2008). *Learning OpenCV*. O'Reilly Media, Inc.
- [Cabanillas, 2009] Cabanillas, F. (2009). Preparan en lepe el prototipo final de un robot para la recogida de fresas.
- [Clavel, 1991] Clavel, R. (1991). *Conception d'un robot parallèle rapide à 4 degrés de liberté*. Docteur thesis es sciences techniques, Ecole Polytechnique Federal de Lausanne (EPFL), Lausanne.
- [Culjak et al., 2012] Culjak, I., Abram, D., Pribanic, T., Dzapo, H., and Cifrek, M. (2012). A brief introduction to OpenCV. In *2012 proceedings of the 35th international convention MIPRO*, pages 1725–1730. IEEE.
- [Cusano, 2022] Cusano, N. (2022). Cobot and Sobot: For a new Ontology of Collaborative and Social Robots. *Foundations of Science*, pages 1–13.
- [De Preter et al., 2018] De Preter, A., Jan Anthonis, and Josse De Baerdemaeker (2018). Development of a robot for harvesting strawberries. *IFAC-PapersOnLine*, 51. 6th IFAC Conference on Bio-Robotics BIOROBOTICS 2018.
- [Dechter, 1986] Dechter, R. (1986). Learning While Searching in Constraint-Satisfaction-Problems. pages 178–185. University of California, Computer Science Department, Cognitive Systems Laboratory. First paper to introduce the term Deep Learning to Machine Learning.
- [Dinamarca, 2018] Dinamarca, A. (2018). Aprendizaje y análisis de redes neuronales artificiales profundas. Tesina de grado, Universidad Nacional de Cuyo. Facultad de Ciencias Exactas y Naturales.
- [Dupont et al., 2021] Dupont, P. E., Nelson, B. J., Goldfarb, M., Hannaford, B., Menchiassi, A., O'Malley, M. K., Simaan, N., Valdastri, P., and Yang, G.-Z. (2021). A decade retrospective of medical robotics research from 2010 to 2020. *Science Robotics*, 6(60).

- [Durmuş et al., 2015] Durmuş, H., Güneş, E. O., Kirci, M., and Üstündağ, B. B. (2015). The design of general purpose autonomous agricultural mobile-robot: “AGROBOT”. In *2015 Fourth International Conference on Agro-Geoinformatics (Agro-geoinformatics)*, pages 49–53.
- [Du-Harpur et al., 2020] Du-Harpur, X., Watt, F., Luscombe, N., and Lynch, M. (2020). What is AI? Applications of artificial intelligence to dermatology. *British Journal of Dermatology*, 183(3):423–430.
- [Díaz-Ramírez, 2021] Díaz-Ramírez, J. (2021). Aprendizaje Automático y Aprendizaje Profundo. *Ingeniare. Revista chilena de ingeniería*, 29(2):180–181.
- [EcoInventos.com, 2024] EcoInventos.com (2024). La primera granja vertical de interior del mundo producirá 1,8 millones de kg de fresas al año.
- [El Naqa and Murphy, 2015] El Naqa, I. and Murphy, M. J. (2015). *What Is Machine Learning?*, pages 3–11. Springer International Publishing.
- [Fernández, 2019] Fernández, M. (2019). El robot español que cosecha las fresas de california sin tocarlas.
- [Fountas et al., 2020] Fountas, S., Mylonas, N., Malounas, I., Rodias, E., Hellmann Santos, C., and Pekkeriet, E. (2020). Agricultural robotics for field operations. *Sensors*, 20(9).
- [GALIPIENSO et al., 2003] GALIPIENSO, A., ISABEL, M., Cazorla Quevedo, M. A., Colomina Pardo, O., Escolano Ruiz, F., and LOZANO ORTEGA, M. A. (2003). *Inteligencia artificial: modelos, técnicas y áreas de aplicación*. Ediciones Paraninfo, SA.
- [García Santillán and Caranqui Sánchez, 2015] García Santillán, I. D. and Caranqui Sánchez, V. M. (2015). La visión artificial y los campos de aplicación. *Tierra Infinita*, 1:98–108.
- [Gasparetto and Scalera, 2019] Gasparetto, A. and Scalera, L. (2019). A Brief History of Industrial Robotics in the 20th Century. *Advances in Historical Studies*, 8:24–35.
- [Gollapudi, 2016] Gollapudi, S. (2016). *Practical machine learning*. Packt Publishing Ltd.
- [Gonzalez-Aguirre et al., 2021] Gonzalez-Aguirre, J. A., Osorio-Oliveros, R., Rodríguez-Hernández, K. L., Lizárraga-Iturrealde, J., Morales Menendez, R.,

- Ramírez-Mendoza, R. A., Ramírez-Moreno, M. A., and Lozoya-Santos, J. d. J. (2021). Service Robots: Trends and Technology. *Applied Sciences*, 11(22):1–22.
- [González and de Mántaras Badia, 2017] González, P. M. and de Mántaras Badia, R. L. (2017). *Inteligencia Artificial*. LOS LIBROS DE LA CATARATA.
- [Gu et al., 2018] Gu, D., Shen, J., and Wang, X. (2018). Deep Learning. *Instructor*.
- [Hardy, 2001] Hardy, T. (2001). IA (Inteligencia Artificial). *Polis: Revista Latinoamericana*, (2):18.
- [ISO/TC299, 2021] ISO/TC299 (2021). *ISO 8373:2021 Robotics — Vocabulary*, pages 1–22. International Organization for Standardization. Only informative sections of standards are publicly available.
- [Janiesch et al., 2021] Janiesch, C., Zschech, P., and Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31:685–695.
- [Kim, 2016] Kim, K. G. (2016). Book Review: Deep Learning. *Healthcare Informatics Research (HIR)*, 22(4):351–354.
- [Koditschek, 2021] Koditschek, D. E. (2021). What Is Robotics? Why Do We Need It and How Can We Get It? *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1):1–33.
- [Kraevsky and Rogatkin, 2010] Kraevsky, S. and Rogatkin, D. (2010). Medical robotics: the first steps of medical robots. *Russian Journal: Technologies of live systems*, 7(4):3–14.
- [Makino and Furuya, 1980] Makino, H. and Furuya, N. (1980). Selective Compliance Assembly Robot Arm. In *Proceedings of 1st International Conference on Assembly Automation (ICAA)*, pages 77–86, Brighton.
- [Martínez Madruga, 2022] Martínez Madruga, J. (2022). Sistema de detección de emociones faciales mediante técnicas de Machine Learning adaptado a ROS para un robot de bajo coste basado en Raspberry Pi. Trabajo de fin de grado, Universidad Rey Juan Carlos.
- [Mokyr et al., 2015] Mokyr, J., Vickers, C., and Ziebarth, N. L. (2015). The History of Technological Anxiety and the Future of Economic Growth: Is This Time Different? *Journal of Economic Perspectives*, 29(3):31–50.

- [Moreno et al., 1994] Moreno, A., Armengol, E., Béjar Alonso, J., Belanche Muñoz, L. A., Cortés García, C. U., Gavaldà Mestre, R., Gimeno, J. M., Martín Muñoz, M., and Sánchez-Marrè, M. (1994). *Aprendizaje automático*. Edicions de la Universitat Politècnica de Catalunya.
- [Oktarina et al., 2020] Oktarina, Y., Dewi, T., Risma, P., and Nawawi, M. (2020). Tomato harvesting arm robot manipulator; a pilot project. *Journal of Physics: Conference Series*, 1500.
- [Ponce Gallegos et al., 2014] Ponce Gallegos, J. C., Torres Soto, A., Quezada Aguilera, F. S., Silva Srock, A., Martínez Flor, E. U., Casali, A., Scheihing, E., Túpac Valdivia, Y. J., Torres Soto, M. D., Ornelas Zapata, F. J., et al. (2014). *Inteligencia artificial*. Iniciativa Latinoamericana de Libros de Texto Abiertos LATIn.
- [Raj and Seamans, 2019] Raj, M. and Seamans, R. (2019). Primer on artificial intelligence and robotics. *Journal of Organization Design*, 8:1–14.
- [Redmon and Farhadi, 2018] Redmon, J. and Farhadi, A. (2018). Yolov3: An Incremental Improvement. *arXiv preprint*, abs/1804.02767.
- [Romero-Tamarit et al., 2020] Romero-Tamarit, A., Reig-Viader, R., Estrada-Sabadell, M. D., and Espallargues-Carreras, M. (2020). Eficacia, efectividad, seguridad y eficiencia de la cirugía robótica con el sistema quirúrgico Da Vinci.
- [Rouhiainen, 2018] Rouhiainen, L. (2018). Inteligencia artificial. *Madrid: Alienta Editorial*, pages 1–22.
- [Sandoval Serrano et al., 2018] Sandoval Serrano, L. J. et al. (2018). Algoritmos de aprendizaje automático para análisis y predicción de datos. *Revista Tecnológica*, no. 11.
- [Schmidhuber, 2015] Schmidhuber, J. (2015). Deep Learning. *Scholarpedia*, 10(11):32832. revision #184887.
- [Siciliano and Khatib, 2016] Siciliano, B. and Khatib, O. (2016). *Industrial Robotics*, pages 963–986. Springer Cham. Text not available. This book is protected by copyright.
- [Siswantoro et al., 2013] Siswantoro, J., Prabuwono, A. S., and Abdullah, A. (2013). Real World Coordinate from Image Coordinate Using Single Calibrated Camera Based on Analytic Geometry. In *Soft Computing Applications and Intelligent Systems*, pages 1–11. Springer Berlin Heidelberg.

- [Suárez et al., 2022] Suárez, R., Rosella, J., Vinagreb, M., Cortesb, F., Ansuateguic, A., Maurtuac, I., Martind, D., Guashd, A., Azpiazuf, J., Serranog, D., et al. (2022). Robot Operating System (ROS). *Este trabajo se ha llevado a cabo por iniciativa y en el seno del Grupo de Trabajo de Innovación de la Asociación Española de Robótica y Automatización (AER)*, 28:1–25.
- [Sánchez Martín et al., 2007a] Sánchez Martín, F., Millán Rodríguez, F., Salvador Bayarri, J., Palou Redorta, J., Rodríguez Escovar, F., Esquena Fernández, S., and Villavicencio Mavrich, H. (2007a). Historia de la robótica: de Arquitas de Tarento al robot Da Vinci (Parte I). *Actas Urológicas Españolas*, 31:69 – 76.
- [Sánchez Martín et al., 2007b] Sánchez Martín, F., Millán Rodríguez, F., Salvador Bayarri, J., Palou Redorta, J., Rodríguez Escovar, F., Esquena Fernández, S., and Villavicencio Mavrich, H. (2007b). Historia de la robótica: de Arquitas de Tarento al Robot Da Vinci. (Parte II). *Actas Urológicas Españolas*, 31:185–196.
- [Vega, 2021] Vega, J. (2021). Basic human–robot interaction system running on an embedded platform. *Microprocessors and Microsystems*, 8(11).
- [Vega and Cañas, 2021] Vega, J. and Cañas, J. M. (2021). Open Vision System for Low-Cost Robotics Education. *Electronics*, 85.
- [Wang et al., 2016] Wang, H., Lei, Z., Zhang, X., Zhou, B., and Peng, J. (2016). Machine learning basics. *Deep learning*, pages 98–164.
- [Wirtz et al., 2018] Wirtz, J., Patterson, P., Kunz, W., Gruber, T., Lu, V., Paluch, S., and Martins, A. (2018). Brave new world: service robots in the frontline. *Journal of Service Management*, 29:1–25.
- [Xiong et al., 2020] Xiong, Y., Ge, Y., Grimstad, L., and From, P. J. (2020). An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. *Journal of Field Robotics*, 37:202–224.
- [Xiong et al., 2019] Xiong, Y., Peng, C., Grimstad, L., From, P. J., and Isler, V. (2019). Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper. *Computers and Electronics in Agriculture*, 157:392–402.
- [Zamalloa et al., 2017] Zamalloa, I., Kojcev, R., Hernández, A., Muguruza, I., Usategui, L., Bilbao, A., and Mayoral, V. (2017). Dissecting Robotics - historical overview and future perspectives.