

Control de velocidad para navegación social inteligente mediante ROS 2

Robinson Jiménez-Moreno¹; Fabio Nicolas Osorio²; Ricardo Castillo³

^{1,3}Universidad Militar Nueva Granada, Colombia, robinson.jimenez@unimilitar.edu.co, ricardo.castillo@unimilitar.edu.co

²Universidad Militar Nueva Granada, Colombia, est.fabio.osorio@unimilitar.edu.co

Resumen– El presente artículo expone el desarrollo de un módulo de visión para desplazamiento robótico basado en el seguimiento de personas, empleando ROS2 para el control de velocidad mediante un nodo publicador embebido en una tarjeta Jetson nano y un nodo suscriptor con un microcontrolador ESP32. De forma que se integra el control de velocidad del móvil acorde a la distancia de detección de una persona mediante una red YOLO, a fin de mantener una distancia segura entre robot y usuario. Se presenta las características de entrenamiento la arquitectura YOLO más reciente basada en versión 12 y se emplea un control PID con un motor D.C. pololu de encoder integrado en el nodo suscriptor, logrando el control de distancia en función a la proximidad del usuario al robot satisfactoriamente.

Palabras clave– ROS2, YOLO, control de velocidad, robot móvil, sistemas embebidos.

I. INTRODUCCIÓN

Los desarrollos basados en robótica y su integración con sistemas operacionales para manipulación de la cinemática inmersa (ROS), gracias a que son de fuente abierta, están direccionando las aplicaciones principalmente en robótica móvil desde enseñanza hasta la investigación [1]. Donde los campos de impacto están a nivel industrial con robot basados en conceptos de industria 4.0 [2], agroclivros con sistemas de visión para navegación [3], líneas de producción para ensamble mediante la programación de tareas y prototipos [4] hasta el diseño de robots para cuidado bajo métodos no intrusivos [5].

ROS también facilita el desarrollo de robots que buscan una interacción natural con el entorno [6] e integración de tareas entre robots [7]. Donde hoy en día se cuenta con una versión mejorada correspondiente a ROS 2, ambos entornos siguen siendo frameworks de desarrollo de software para robótica, pero ROS2 representa una evolución significativa con diferencias clave que mejoran la arquitectura, el rendimiento, la compatibilidad con entornos industriales y las características de seguridad. Su uso permite el control [8][9] y la supervisión de los recursos del robot [10][11].

Las capacidades de los sistemas robóticos se están incrementando con sistemas de visión basados en inteligencia artificial [12] y plataformas de hardware móvil con sistemas embebidos [13], mejorando la percepción del robot [14] incluso con interacción por lenguaje natural [15]. Donde dicha interacción emplea algoritmos como arquitecturas de aprendizaje profundo como el modelo YOLO para sistemas de visión [16] en tareas tan específicas como agarre de objetos [17]. De forma que las variantes de esta arquitectura para detección y localización de objetos en imágenes [18] se evalúan permanentemente [19] y han derivado en modelos ligeros aptos para sistemas embebidos y de uso remoto [20].

Para plataformas robóticas implementación de ROS se ha realizado mediante sistemas embebidos como las tarjetas Jetson de Nvidia, El cual puede operar de forma eficiente incluso en plataformas robóticas aéreas como UAV [21]. De igual forma la implementación de algoritmos de aprendizaje profundo dentro de las plataformas Jetson también ha demostrado un eficiente desempeño en diferentes tipos de plataformas robóticas como robots móviles y drones [22]. Donde en [23] se puede evidenciar un desarrollo en el cual se utiliza la versión cuatro de la red YOLO para la detección de peatones desde el sistema embebido Jetson con el objeto de obtener un flujo de información preciso y en tiempo real de la localización de peatones.

Integrando los beneficios de ROS2 y la última versión de la red YOLO al momento del desarrollo, el presente artículo expone el desarrollo de un módulo de visión para desplazamiento robótico basado en el seguimiento de personas, embebido en una tarjeta Jetson nano. De forma que se integra el control de velocidad del móvil acorde a la distancia de detección de la persona a fin de mantener una distancia segura entre robot y usuario.

El presente artículo consta entonces de cuatro secciones la actual introducción donde se expone brevemente el estado del arte en el tema de navegación robótica social. La sección 2 que establece la metodología que se emplea la cual se resume mediante un diagrama de flujo. La sección 3 en la que se expone el análisis de resultados de la integración el robot con el sistema embebido Jetson y los algoritmos implementados. Finalmente se presenta la sección cuatro con las conclusiones alcanzadas.

II. METODOLOGIA

La metodología propuesta corresponde a una investigación aplicada derivada del enfoque a solucionar un problema específico correspondiente a evitar la colisión del robot con una persona, requiriendo que este la identifique en su entorno. Se exponen tres fases en el desarrollo metodológico, primero el entrenamiento y uso de la versión YOLO más reciente a la fecha de elaboración del documento para el caso la versión 12. Segundo la implementación en ROS y tercero la validación en la integración de las dos primeras fases para el control de velocidad de un móvil robótico, empleando un control PID con un motor D.C. Pololu de encoder integrado.

La figura 1 ilustra el proceso general que se emplea después de realizar cada una de las fases metodológicas. Una vez configurado ROS en la tarjeta Jetson se procede a entrenar mediante ajuste fino la red YOLO, la cual si presenta una precisión mayor al 90% se considera como válida, con la red

funcional se hacen pruebas con el motor después de la conexión con la tarjeta con la ESP32 que lo controlará, finalmente si se le valida el seguimiento de la persona y el robot se detiene el proceso finaliza.

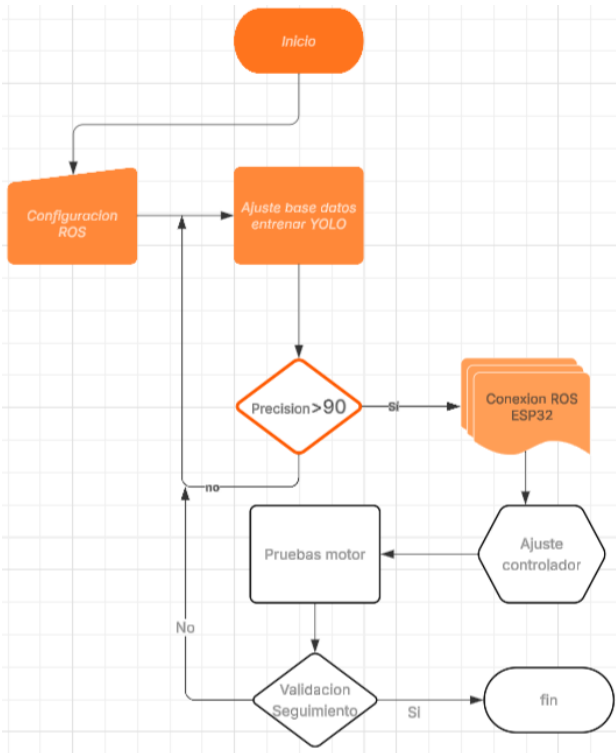


Fig. 1 Diagrama de flujo del proceso

A. Fase I

La red YOLOv12 emplea una arquitectura de atención optimizada, es decir, con relación a modelos previos optimiza el mecanismo de atención estándar para una mayor eficiencia y compatibilidad, usando FlashAttention para minimizar la sobrecarga de acceso a la memoria, eliminando la codificación posicional y ajustando la relación multicapa para equilibrar mejor el cálculo entre la atención y las capas de alimentación hacia adelante. Esto permite la reducción de la profundidad de los bloques apilados y un mejor aprovechamiento de las operaciones de convolución por su eficiencia computacional [24].

Para el entrenamiento se emplea una base de datos en entorno cerrado como se aprecia en la Fig. 2, donde el entrenamiento se orienta a identificar elementos del lugar que se diferencien del usuario. A la vez se presentan ejemplos de persona muy cerca y con oclusiones parciales para mejorar la identificación.



Fig. 2 Muestra base de datos de entrenamiento YOLO v12. Fuente propio

La Fig. 3 ilustra las gráficas del entrenamiento de la red donde se evidencia que el nivel de precisión y recall es superior al 95%, con muy pocas confusiones entre clases, denotando el buen desempeño de la arquitectura.

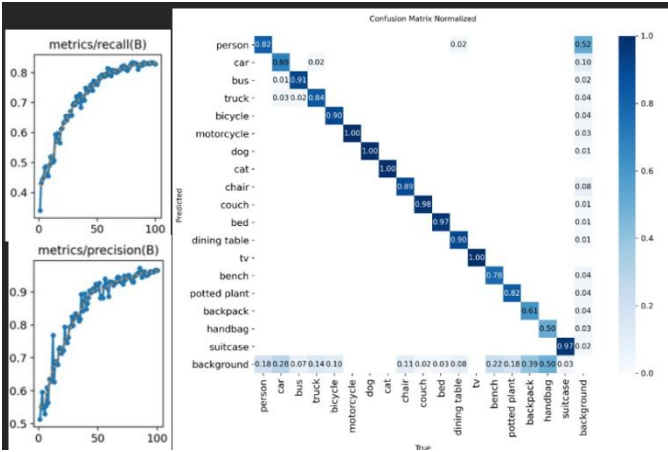


Fig. 3 Métricas de desempeño del entrenamiento y Matriz de confusión YOLOv12. Fuente propio

La Fig. 4 ilustra el resultado de identificación con oclusiones, donde los niveles de confianza están por encima de 80% en cada clase, pero para una oclusión que divide el objeto a identificar (hombre al fondo) no logra el reconocimiento.

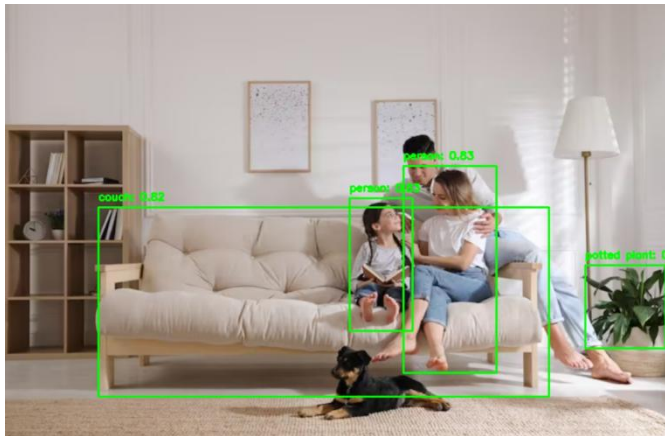


Fig. 4 Resultados de la red en el entrenamiento YOLO v12. Fuente propio

B. Fase 2

ROS2 es usado para el control robótico mediante un nodo publicador embebido en una tarjeta Jetson nano de NVIDIA y un nodo suscriptor [26] con un microcontrolador ESP32, como se aprecia en la Fig. 5. La tarjeta Jetson corre ROS para el control general del móvil robótico y el reconocimiento por cámara de usuario mediante la red YOLO, a su vez se interconecta con la ESP32 para comando del motor DC.

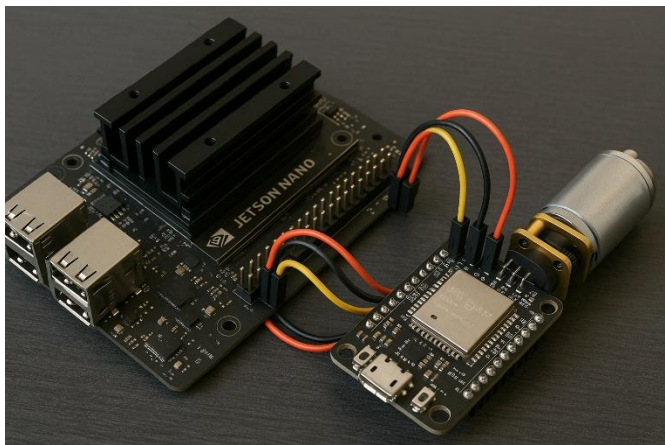


Fig. 5 Jetson Nano y ESP32 para microROS.

C. Fase 3

En la fase 3 se evalúa de forma independiente la variación de la velocidad desde ROS donde la figura 6 ilustra la curva de respuesta del motor en lazo abierto ante un escalón, lo que permite establecer las características de diseño de un módulo de control.

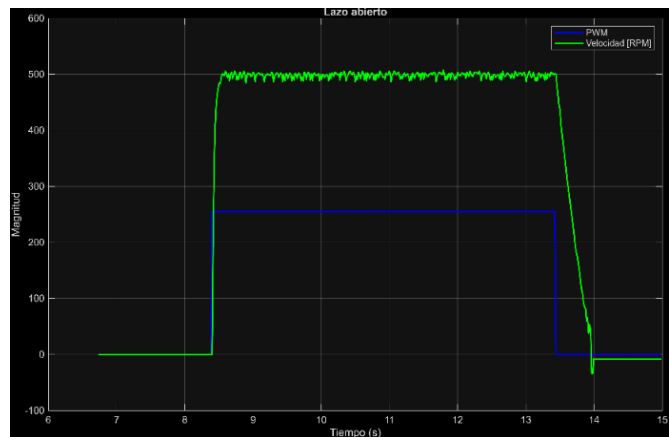


Fig. 6 Respuesta del sistema en lazo abierto.

Se pueden observar en (1) a (3), las constantes del modelo de control establecido en (4), parten de la discretización de un control PID continuo por el método de tustin [25], para la regulación de la velocidad. La ecuación (4) modifica el ancho de pulso que es enviado al motor.

$$a0 = Kp(1 + Tm/2Ti + Td/Tm) \quad (1)$$

$$a1 = Kp(Tm/2Ti - 2 \cdot Td/Tm - 1) \quad (2)$$

$$a2 = Kp \cdot Td/Tm \quad (3)$$

$$u[k] = u[k - 1] + a0 \cdot e[k] + a1 \cdot e[k - 1] + a2 \cdot e[k - 2] \quad (4)$$

Estableciendo los parámetros de diseño del control para un T_s de 0.2s y un sobre impulso inferior al 30% se obtuvieron las siguientes constantes para el control proporcional integral, derivativo (PID):

$$Kp = 0.372$$

$$Ti = 0.165$$

$$Td = 0.022$$

$$Tm = 0.015$$

En la Fig. 7 se evidencia la respuesta de control recibiendo la referencia por comandos desde el nodo publicador en la Jetson nano, con una respuesta sin error en estado estacionario y mínimo sobreimpulso.

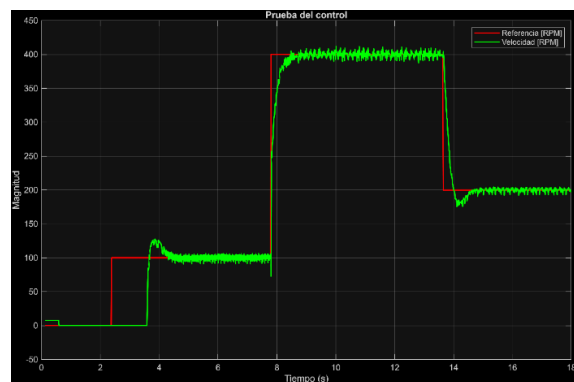


Fig. 7 Control velocidad motor mediante ROS2.

III. ANÁLISIS DE RESULTADOS

Las Fig. 8 a 10 ilustran el resultado del seguimiento de la persona mediante la red YOLO a diferentes distancias y poses respecto a la cámara, obteniendo niveles de confianza en promedio por encima del 85%, lo que permite la identificación y seguimiento adecuado. La Figura 8 ilustra la detección de frente de persona, Donde se puede observar que el nivel de conciencia en la detección es bastante alto para el caso 96%.



Fig. 8 Detección de la personal de frente.

En la figura 9 se puede observar que la detección de la persona de espaldas también presenta niveles de confianza alto cuándo se encuentra cerca de la cámara. Sin embargo, en la figura 10 se puede apreciar que se da la detección satisfactoria de la persona, aunque el nivel de confianza bajó al 86% lo cual establece un rango de cobertura adecuado para la detección de la persona de forma tal que para asegurar el funcionamiento del algoritmo se coloca un umbral de detección el 75%. Esto permite asegurar que el robot identificará de forma acertada a una persona entonces de un radio que le permita tomar acción preventiva para no chocar con ella.



Fig. 9 Detección de la persona de espaldas.



Fig. 10 detección de frente lejos del robot.

La Fig. 11. presenta la comparativa de las señales de mayor interés para el control del motor por seguimiento de usuario. Se toma como referencia el tamaño del recuadro de detección que es proporcional a la distancia (línea verde), véase que cuando el tamaño aumenta o sea el individuo se encuentra más cerca, la referencia del control disminuye (línea roja) logrando así una menor velocidad de giro (línea azul), tiempo hasta los 3 segundos y de 9 a los 11 segundos. También se presenta el caso contrario donde el individuo se aleja por lo que tamaño del recuadro disminuye y la referencia aumenta junto con la velocidad del motor, tiempo desde los 4 hasta los 8 segundos y de 12 a los 15 segundos.

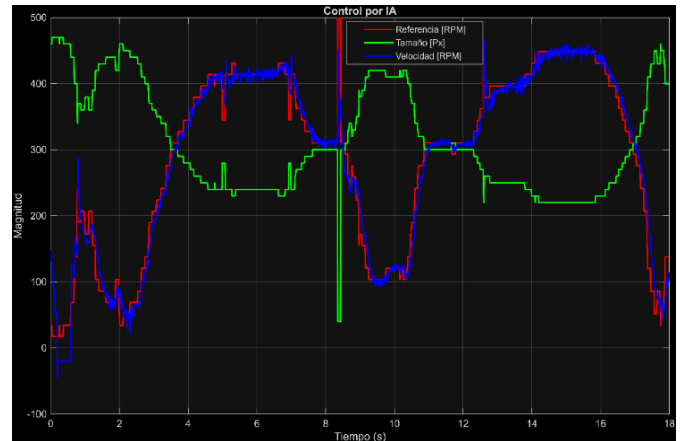


Fig. 11 Control por seguimiento mediante ROS2 y YOLO.

La razón de conversión del recuadro basado en su tamaño en pixeles de entrada a una referencia de velocidad en RPMs para el control del motor parte de una interpolación lineal aplicada con programación mediante (5).

$$Ref_{in} = Ref_{Min} + (Tamaño_{in} - Tamaño_{Min}) * \frac{(Ref_{Max} - Ref_{Min})}{(Tamaño_{Max} - Tamaño_{Min})} \quad (5)$$

IV. CONCLUSIONES

Se logra evidenciar que la integración de ROS y sistemas de detección por visión mediante red YOLO permiten una respuesta del motor en tiempo real y así una reacción oportuna del sistema de control de velocidad. La rápida respuesta obtenida de los cambios de velocidad frente a la proximidad del usuario permite concluir que el desarrollo es aplicable a modelos de robótica asistencial que eviten colisiones humano-robot.

Se puede concluir que el uso de algoritmos de inteligencia artificial dentro de modelos Jetson nano logra cubrir los requerimientos de movilidad robótica necesaria para aplicaciones en entornos sociales que le permitan al robot mantener una interacción segura dentro del espacio desplazamiento con las personas que es circundan este.

Cómo trabajo futuro se establece el lograr determinar posibles variaciones de velocidad para desplazamiento del robot de forma proporcional a la distancia en que se encuentre el usuario. A su vez se tiene el probar el algoritmo bajo la detección de múltiples personas en el entorno.

AGRADECIMIENTOS

Los autores agradecen a la Universidad Militar Nueva Granada de donde son docente de planta y a la vicerrectoría de investigaciones por la financiación del proyecto titulado “Diseño de un agente inteligente móvil de uso residencial para el cuidado básico de personas, basado en un modelo de descripción visual” código IMP-ING-4307 para el año 2026.

REFERENCIAS

- [1] Renno Raudmäe, Sandra Schumann, Veiko Vunder, Maarika Oidekivi, Madis Kaspar Nigol, Robert Valner, Houtman Masnavi, Arun Kumar Singh, Alvo Aabloo, Karl Kruusamäe, ROBOTONT – Open-source and ROS-supported omnidirectional mobile robot for education and research, *HardwareX*, Volume 14, 2023, e00436, ISSN 2468-0672, <https://doi.org/10.1016/j.ohx.2023.e00436>.
- [2] Houssam Bazza, Sandro Bimonte, Zakaria Gourt, Stefano Rizzi, Hassan Badir, An MDA approach for robotic-based real-time business intelligence applications, *Data & Knowledge Engineering*, Volume 157, 2025, 102418, ISSN 0169-023X, <https://doi.org/10.1016/j.datak.2025.102418>.
- [3] Abdelkrim Abanay, Lhoussaine Masmoudi, Dirar Benkhedra, Khalid El Amraoui, Mouataz Lghoul, Javier-Gonzalez Jimenez, Francisco-Angel Moreno, A transformation model for vision-based navigation of agricultural robots, *Cognitive Robotics*, Volume 5, 2025, Pages 140-151, ISSN 2667-2413, <https://doi.org/10.1016/j.cogr.2025.03.002>.
- [4] Niklas Terei, Rolf Wiemann, Annika Raatz, ROS-Based Control of an Industrial Micro-Assembly Robot, *Procedia CIRP*, Volume 130, 2024, Pages 909-914, ISSN 2212-8271, <https://doi.org/10.1016/j.procir.2024.10.184>.
- [5] Ruomeng Zhang, Solan Keishing, Jims Marchang, Raymond Mawanda, Ning Wang, Alessandro Di Nuovo, Non-intrusive continuous user verification by care robots: MoveNet gait data, *Intelligent Sports and Health*, Volume 1, Issue 3, 2025, Pages 160-178, ISSN 3050-5445, <https://doi.org/10.1016/j.ish.2025.06.003>.
- [6] Sara Bouraine, Yassine Bellalia, Ilyes Chaabeni, Djamila Naceur, When robots learn from nature: GLWOA-RR*, a nature-inspired motion planning approach, *Swarm and Evolutionary Computation*, Volume 98, 2025, 102062, ISSN 2210-6502, <https://doi.org/10.1016/j.swevo.2025.102062>.
- [7] Francisco José Martínez-Peral, Jorge Borrell Méndez, Dennis Mronga, José Vicente Segura-Heras, Carlos Perez-Vidal, Trajectory planning system for bimanual robots: Achieving efficient collision-free manipulation, *Robotics and Autonomous Systems*, Volume 194, 2025, 105118, ISSN 0921-8890, <https://doi.org/10.1016/j.robot.2025.105118>.
- [8] Walter Fetter Lages, Joint Level Adaptive Control for Mobile Robots in ROS 2, *IFAC-PapersOnLine*, Volume 55, Issue 38, 2022, Pages 172-177, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2023.01.151>.
- [9] Walter Fetter Lages, Remote Teaching of Dynamics and Control of Robots Using ROS 2, *IFAC-PapersOnLine*, Volume 55, Issue 17, 2022, Pages 279-284, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2022.09.292>.
- [10] Giovanni Stanco, Alessio Botta, Luigi Gallo, Giorgio Ventre, DewROS2: A platform for informed Dew Robotics in ROS, *Robotics and Autonomous Systems*, Volume 182, 2024, 104800, ISSN 0921-8890, <https://doi.org/10.1016/j.robot.2024.104800>.
- [11] Steve Macenski, Tom Moore, David V. Lu, Alexey Merzlyakov, Michael Ferguson, From the desks of ROS maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2, *Robotics and Autonomous Systems*, Volume 168, 2023, 104493, ISSN 0921-8890, <https://doi.org/10.1016/j.robot.2023.104493>.
- [12] Khushniddin Fozilov, Yutaro Yamada, Jacinto Colan, Yaonan Zhu, Yasuhisa Hasegawa, IST-ROS: A flexible object segmentation and tracking framework for robotics applications, *SoftwareX*, Volume 29, 2025, 101979, ISSN 2352-7110, <https://doi.org/10.1016/j.softx.2024.101979>.
- [13] Rubén Nieto, Felipe Machado, Jesús Fernández-Conde, David Lobato, José M. Cañas, Open-source ROS-based simulation for verification of FPGA robotics applications, *Microprocessors and Microsystems*, Volume 113, 2025, 105143, ISSN 0141-9331, <https://doi.org/10.1016/j.micpro.2025.105143>.
- [14] Tomáš Zemčík, Lukáš Kratochvíla, Mobile robot perception system in ROS 2 on embedded computing platforms, *IFAC-PapersOnLine*, Volume 58, Issue 9, 2024, Pages 305-310, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2024.07.414>.
- [15] Bilel Benjdira, Anis Koubaa, Anas M. Ali, Prompting Robotic Modalities (PRM): A structured architecture for centralizing language models in complex systems, *Future Generation Computer Systems*, Volume 166, 2025, 107723, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2025.107723>.
- [16] S. -H. Wu and X. -S. Hong, "Integrating Computer Vision and Natural Language Instruction for Collaborative Robot Human-Robot Interaction," 2020 International Automatic Control Conference (CACs), Hsinchu, Taiwan, 2020, pp. 1-5, doi: 10.1109/CACs50047.2020.9289768.
- [17] O. Caixiao, "Research on Object Grasping System Based on YOLO Algorithm," 2024 International Conference on Telecommunications and Power Electronics (TELEPE), Frankfurt, Germany, 2024, pp. 237-241, doi: 10.1109/TELEPE64216.2024.00048.
- [18] Muhammad Adil Raja, Róisín Loughran, Fergal McCaffery, A Review of Performance of Recent YOLO Models on Cholecystectomy Tool Detection, *Measurement: Digitalization*, Volumes 2–3, 2025, 100007, ISSN 3050-6441, <https://doi.org/10.1016/j.measdig.2025.100007>.
- [19] Leila Ismail, Nada Shahin, Henok Tesfaye, Alain Hennebelle, VisioSLR: A Vision Data-Driven Framework for Sign Language Video Recognition and Performance Evaluation on Fine-Tuned YOLO Models, *Procedia Computer Science*, Volume 257, 2025, Pages 85-92, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2025.03.014>.
- [20] Xuechen Bai, Xinghua Li, Jianhao Miao, Huanfeng Shen, A front-back view fusion strategy and a novel dataset for super tiny object detection in remote sensing imagery, *Knowledge-Based Systems*, Volume 326, 2025, 114051, ISSN 0950-7051, <https://doi.org/10.1016/j.knosys.2025.114051>.
- [21] X. Wang, Z. Zhou e Y. Li, "Diseño de un sistema de seguimiento de objetivos en movimiento basado en la plataforma Jetson ", Conferencia Internacional IEEE sobre Inteligencia Artificial y Sistemas de Información (ICAIS) de 2020 , Dalian, China, 2020, págs. 371-375, doi: 10.1109/ICAIS49377.2020.9194918.
- [22] M. Choe, S. Lee, N.-M. Sung, S. Jung y C. Choe, "Análisis comparativo de detectores de objetos 3D basados en aprendizaje profundo en plataformas NVIDIA Jetson ", Conferencia Internacional sobre Convergencia de Tecnologías de la Información y la Comunicación (ICTC) de 2021, Isla de Jeju, Corea, República de, 2021, págs. 10-12, doi: 10.1109/ICTC52510.2021.9621072.

- [23]Y. Wang, R. Zou, Y. Chen y Z. Gao, "Investigación sobre detección de peatones basada en la plataforma Jetson Xavier NX y YOLOv4", 4.º Simposio Internacional sobre Ingeniería Informática y Comunicaciones Inteligentes (ISCEIC) , Nanjing, China, 2023, págs. 373-377, doi: 10.1109/ISCEIC59030.2023.10271216.
- [24]<https://docs.ultralytics.com/es/models/yolo12/#key-features> , consultado en Julio 30 de 2025.
- [25]Franklin, G.F., Powell, D.J., and Workman, M.L., Digital Control of Dynamic Systems (3rd Edition), Prentice Hall, 1997.
- [26]S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall, "Robot Operating System 2: Design, architecture, and uses in the wild," Science Robotics, vol. 7, no. 66, May 2022, doi: 10.1126/scirobotics.abm6074