

Navegación autónoma en interiores basada en localización visual Autonomous indoor navigation based on visual location

H. I. Rico-Mendoza ^{a,*}, M. Reyna-Rodríguez ^a, A. B. Morales-Díaz ^a, K. Ordaz-Hernández ^a, C. Treesatayapun ^a

^aGrupo de Robótica y Manufactura Avanzada, Centro de Investigación y de Estudios Avanzados del IPN-Salttillo, Industrial, Zona Industrial, 25900 Ramos Arizpe, Coahuila, México.

Resumen

En este artículo se desarrolla una ley de control para navegación autónoma en interiores basada en la detección visual de marcadores de referencia. Técnicas de estimación de pose externas como la localización por GPS o sensores RGB-D en techos son complicados de implementar en entornos cerrados donde puede haber obstrucción de vista o señal, como en almacenes, por lo que la estimación de pose local con sensores a bordo presenta una solución más viable. La implementación de cámaras web de alta definición supone una solución más económica que el uso de sensores de alta calidad como sensores láser tipo Lidar. En la ley de control diseñada se considera a un marcador visual ArUco en su campo de visión, como un marco referencial inercial local. Con base en los errores medidos por odometría es posible ejecutar la tarea de regulación hacia éste marcador.

Palabras Clave: ArUco, Navegación autónoma, Navegación visual, Robot móvil.

Abstract

In this article, a control law for autonomous indoor navigation based on the visual detection of fiducial markers is developed. External pose estimation techniques such as GPS tracking or RGB-D sensors on ceilings are difficult to implement in closed environments where there may be view or signal obstruction, such as in warehouses, so local pose estimation with on-board sensors presents a better solution. The implementation of high-definition webcams is a cheaper option than the use of high-quality sensors such as laser ones (i.e. Lidar). In the designed control law, an ArUco visual marker is considered in a robot field of vision, as a local inertial frame of reference. Based on the errors measured by odometry, it is possible to execute the regulation task towards this marker.

Keywords: ArUco, Autonomous navigation, Visual navigation, Mobile robot.

1. Introducción

La navegación de robots móviles en interiores presenta dificultades como la imposibilidad de usar GPS para determinar la posición del robot, así como la dificultad de establecer trayectorias de navegación en ambientes estructurados complejos. Algunas soluciones a estas problemáticas se han planteado usando sensores tipo Lidar, ultrasónicos, etc (Zhang *et al.*, 2021).

Sin embargo, es común que al usar algunos de los sensores mencionados se presente un error acumulable (Guo *et al.*, 2020); por esto se apuesta cada vez más a opciones como la navegación a través de detección visual.

La accesibilidad a cámaras de alta resolución en contraste a los precios de sensores de alto rendimiento ha llevado a su uso más pronunciado. Aunado a lo anterior, se tiene que con el desarrollo de técnicas de visión por computadora, es factible hoy en día realizar una navegación autónoma mediante marcas de referencias y obtener la estimación de su pose y orientación (de Oliveira Junior *et al.*, 2021; Babinec *et al.*, 2014).

2. Navegación basada en detección visual

Un marcador de referencia es un sistema visual donde una señal gráfica es colocada en el entorno, y con ayuda de técnicas de visión se puede determinar la distancia y orientación entre la

*Autor para correspondencia: ivan.rico@cinvestav.mx

Correo electrónico: ivan.rico@cinvestav.mx (Hugo Iván Rico-Mendoza), misael.reyna@cinvestav.mx (Misael Reyna-Rodríguez), america.morales@cinvestav.edu.mx (América Berenice Morales-Díaz), kenny.diaz@cinvestav.edu.mx (Keny Ordaz-Hernández), chidentree@cinvestav.edu.mx (Chidentree Treesatayapun).

cámara y el marcador. Existen muchas marcas utilizables tales como ARTag, ARToolkit, AprilTag, ArUco, STag (de Oliveira Junior *et al.*, 2021).

Los marcadores de referencia cuadrados pueden ser detectados y ubicados si sus casillas de marcado son de al menos el tamaño de un píxel, lo que puede ser detectado a grandes distancias, dependiendo del tamaño real del marcador (OpenCV, 2022).

Los marcadores ArUco son imágenes sintéticas cuadradas que tienen un borde negro que facilita la detección de la matriz interna binaria como se muestra en la Figura 1. La configuración única obtenida en la matriz interna binaria determina el identificador de cada ArUco dentro del diccionario; entre más grande sea la matriz interna, hay más posibilidad de que exista un número mayor de marcadores en el diccionario, sin embargo, es posible seleccionar la cantidad necesaria para disminuir el tamaño de éste (OpenCV, 2022).

El proceso de localización de OpenCV en Python para los ArUco, como se muestra en la Figura 1, ofrece una alternativa fuerte para la ubicación de la cámara con respecto a un marco referencia local, lo cual ofrece una alternativa para realizar tareas de navegación autónoma (Poroykov *et al.*, 2020).

3. Navegación autónoma

La navegación presentada en este trabajo busca la autonomía a partir de la identificación de marcadores referenciales ArUco, para poder estimar la pose del robot móvil, y con la cual se calculan las señales de control para las ruedas.

3.1. Modelo

El robot móvil utilizado en este artículo puede ser modelado como uno de tipo unicyclo. En la Figura 2, se puede observar el diagrama de este robot.

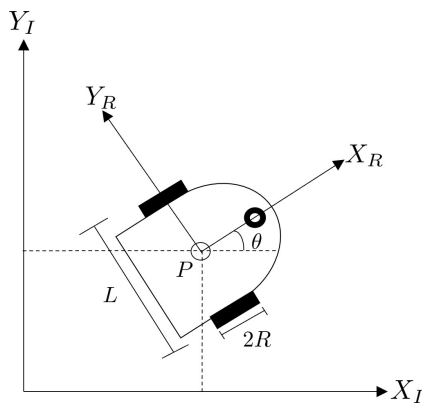


Figura 2: Robot móvil unicyclo.

El movimiento del robot móvil se encuentra sujeto a una restricción de no holonomía debido al no deslizamiento lateral de sus llantas. Las restricciones cinemáticas usualmente son representadas de la forma *Pfaffian* (De Luca *et al.*, 2001)

$$A(q)\dot{q} = 0.$$

Siendo las coordenadas generalizadas $q = (x, y, \theta)$ para este tipo de robot; entonces, la restricción no holónoma de las llantas es representada por:

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0;$$

la cual expresada en la forma *Pfaffian* es:

$$A(\mathbf{q})\dot{\mathbf{q}} = \begin{bmatrix} \sin \theta & -\cos \theta & 0 \end{bmatrix} \begin{bmatrix} \dot{x} & \dot{y} & \dot{\theta} \end{bmatrix}^T = 0.$$

Considerando la siguiente matriz $G(\mathbf{q})$, cuyas columnas $\mathbf{g}_1(\mathbf{q})$ y $\mathbf{g}_2(\mathbf{q})$ son bases del espacio nulo de $A(\mathbf{q})$

$$G(\mathbf{q}) = \begin{bmatrix} \mathbf{g}_1(\mathbf{q}) & \mathbf{g}_2(\mathbf{q}) \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix};$$

y además considerando que todas las velocidades generalizadas admisibles para el sistema, v velocidad del robot en dirección de X_R según la Figura 2 y ω la velocidad angular sobre el punto P del robot mostrado en la Figura 2, se obtienen a través de una combinación lineal de $\mathbf{g}_1(\mathbf{q})$ y $\mathbf{g}_2(\mathbf{q})$, entonces el modelo cinemático es:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = g_1(q)v + g_2(q)\omega = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega. \quad (1)$$

Es posible describir el comportamiento cinemático del robot en función de las velocidades angulares de rueda izquierda ω_l y rueda derecha ω_r . La velocidad lineal se describe como el promedio de las velocidades angulares de cada rueda:

$$v = \frac{R}{2}(\omega_R + \omega_L). \quad (2)$$

Por otra parte, la velocidad angular se define como la relación de la velocidad angular de las ruedas y el radio de rotación:

$$\omega = \frac{R}{L}(\omega_R - \omega_L); \quad (3)$$

donde R representa el radio de las llantas y L representa la distancia entre éstas. Dado que las ruedas están sujetas a una restricción no holónoma, la velocidad lateral es siempre cero; entonces, la relación entre las ω_R y ω_L , y v y ω se calcula como:

$$\begin{bmatrix} v \\ v_{\perp} \\ \omega \end{bmatrix} = \begin{bmatrix} R/2 & R/2 \\ 0 & 0 \\ R/L & -R/L \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix},$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} R/2 & R/2 \\ R/L & -R/L \end{bmatrix} \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix},$$

por lo tanto,

$$\begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} = \frac{1}{2R} \begin{bmatrix} 2 & L \\ 2 & -L \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (4)$$

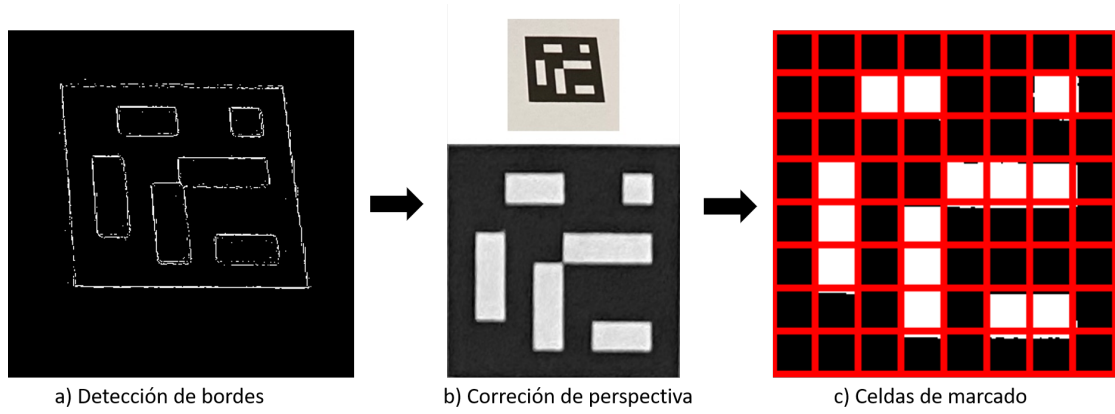


Figura 1: Línea de trabajo para la identificación de marcadores ArUco con su biblioteca de funciones en OpenCV (OpenCV, 2022); a) la imagen es sometida a una identificación de bordes para detectar el patrón de algún posible marcador, b) una vez identificado se realizan las transformaciones elementales necesarias para colocar la imagen paralela al plano de la cámara y c) finalmente se realiza un marcado de casillas.

3.2. Ley de control

El control de regulación fue diseñado con base en el control de seguimiento de trayectoria presentado en el artículo (Gutiérrez et al., 2017).

Los errores con respecto al referencial inercial se en termino del punto deseado (X_D, Y_D) y la posición actual del robot (X_R, Y_R) definen como:

$$e_I = \begin{bmatrix} e_x \\ e_y \end{bmatrix} = \begin{bmatrix} X_D - X_R \\ Y_D - Y_R \end{bmatrix}$$

$$e_\theta = \theta_d - \theta,$$

por lo tanto, las coordenadas del error con respecto al marco local están dadas por:

$$e = [e_x, e_y]^T = \mathbf{R}^T(\theta)e_I \quad (5)$$

donde

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.$$

De (5) la dinámica del error se representa de la siguiente manera:

$$\dot{e}_x = \omega e_y - v, \quad (6)$$

$$\dot{e}_y = -\omega e_x, \quad (7)$$

$$\dot{e}_\theta = -\dot{\theta} = -\omega. \quad (8)$$

Con base en (6), (7) y (8), se proponen las siguientes entradas de control:

$$v = k_x e_x \quad (9)$$

$$\omega = \frac{\sin(e_\theta)}{e_\theta} (k_y e_y) + k_\theta e_\theta \quad (10)$$

Con el fin de probar la estabilidad del sistema bajo en control propuesto (9),(10); se propone la siguiente función definida positiva:

$$V(e) = \frac{1}{2}(e_x^2 + e_y^2 + e_\theta^2),$$

cuya derivada con respecto al tiempo es:

$$\dot{V}(e) = e_x \dot{e}_x + e_y \dot{e}_y + e_\theta \dot{e}_\theta, \quad (11)$$

Sustituyendo (6), (7) y (8) en (11) se tiene:

$$\begin{aligned} \dot{V}(e) &= e_x(\omega e_y - v) + e_y(-\omega e_x) - e_\theta \omega, \\ &= -v e_x - \omega e_\theta, \end{aligned}$$

y sustituyendo las entradas de control v y ω se tiene:

$$\begin{aligned} \dot{V}(e) &= -k_x e_x^2 - \sin(e_\theta) k_y e_y - k_\theta e_\theta^2, \\ &= -k_x e_x^2 - \gamma k_y e_y^2 - k_\theta e_\theta^2 + \gamma k_y e_y^2 - \sin(e_\theta) k_y e_y, \\ &\leq -k_x e_x^2 - \gamma k_y e_y^2 - k_\theta e_\theta^2 + \gamma k_y e_y^2 - k_y e_y; \end{aligned} \quad (12)$$

donde γ es un valor real positivo arbitrario, si $e_y \geq \frac{1}{\gamma}$ sucede que $k_y e_y^2 - k_y e_y = 0$ y $-k_x e_x^2 - \gamma k_y e_y^2 - k_\theta e_\theta^2 \leq 0$ por lo tanto, de (12) se obtiene que $\dot{V}(e) \leq 0$, dado que no hay alguna otra restricción sobre γ , si ésta tiende a infinito, el error $e_y \rightarrow 0$, por lo que el control es estable en una cota (Khalil, 2015).

3.3. Control visual

Como se muestra en la Figura 3, la detección del ArUco y las transformaciones elementales para la estimación de la cámara con respecto a este, son necesarias para la estimación completa de la pose local del robot móvil.

El control visual tiene como objetivo localizar el marcador de referencia, procesar la información que éste provee y obtener los errores locales del robot para aplicar el control de regulación (9), (10) hacia este marcador.

De acuerdo a la Figura ??, al detectar el marcador en el campo de visión de la cámara a bordo del robot móvil (Figura 6) se utiliza la librería ArUco de OpenCv para detectar la coordenada del error en el eje x , e_x en centímetros del objetivo con respecto al marco del robot, después se obtiene la coordenada del error

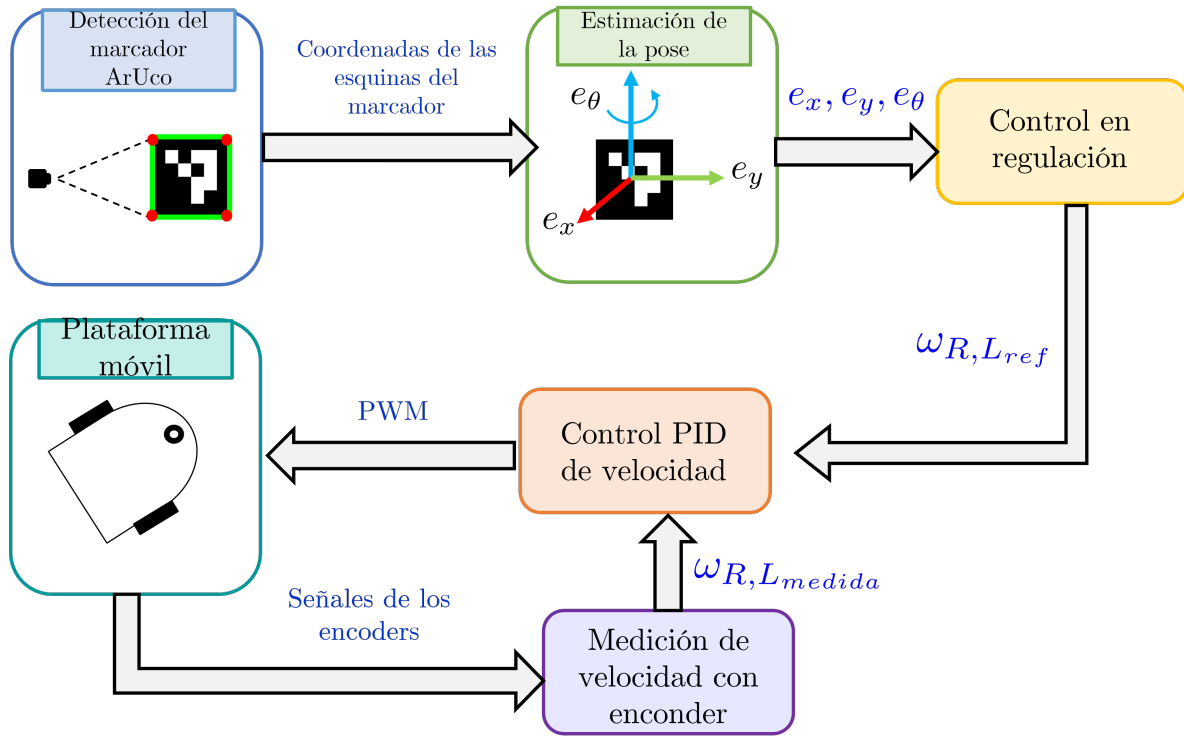


Figura 3: Diagrama de funcionamiento para el control de navegación autónoma basada en localización visual, nótese que se utiliza la notación $\omega_{R,Lref}$ para las velocidades angulares para ruedas calculadas por el control, y $\omega_{R,Lmedida}$ para las velocidades angulares calculadas a partir de los encoders.

en y , e_y , del objetivo respecto referencial del robot mediante la siguiente ecuación:

$$e_y = e_x \tan(e_\theta). \quad (13)$$

Para obtener el error en orientación del objetivo con respecto al robot se hace uso de la siguiente ecuación:

$$e_\theta = \frac{\pi}{3(W)} \left(\frac{W}{2} - C \right), \quad (14)$$

donde:

- W es el ancho de la imagen en píxeles.
- C es la coordenada horizontal del centro del marcador ArUco.
- $\frac{\pi}{3}$ representa el campo de visión de la cámara.

Los errores e_x , e_y , e_θ coinciden con los errores calculados con respecto al marco local en (6), (7) y (8), por lo tanto es válido usar los errores de esta sección en la ley de control (9) y (10).

Debido a las limitantes físicas del campo de visión y la máxima profundidad que permite la cámara, es suficiente que el robot se regule al marco referencial local que se determina por el ArUco.

4. Resultados

El marco experimental consta de un ArUco colocado de manera paralela al plano de la cámara como se muestra en la Figura 5. El robot móvil consta de dos motores con *encoders*

de 800 pulsos por revolución, los cuales están conectados a un Arduino Uno; este a su vez conectado por puerto serial a una computadora Odroid (Odroid, 2022) a bordo con un sistema operativo Ubuntu 8, Opencv 4.10 y Python3, donde se realiza el procesamiento de la imagen, el cálculo del control y el envío de la señal al Arduino Uno (Arduino, 2022).

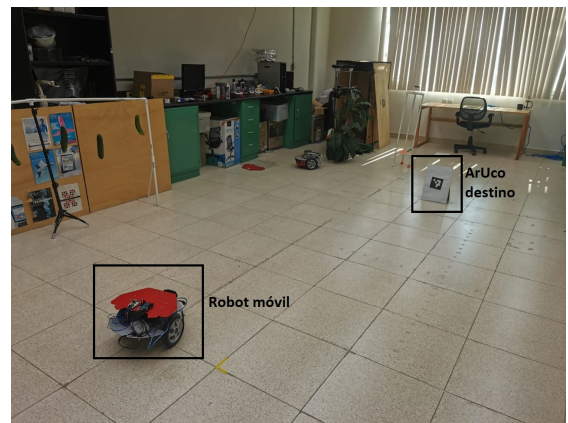


Figura 5: El experimento fue llevado a cabo dentro de las instalaciones de Grupo de Robótica y Manufactura Avanzada del Cinvestav Saltillo

Se muestran los resultados de colocar el robot en un desfase de $\Delta_x = 300cm$ y $\Delta_y = 200cm$, con ganancias $k_x = 1$, $k_y = 5 \times 10^{-3}$ y $k_\theta = 15$.

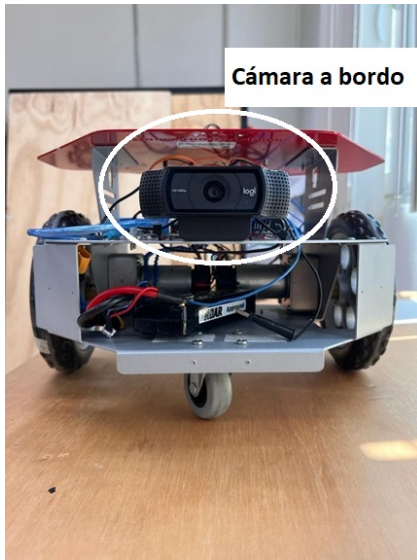


Figura 6: El robot móvil funciona con una cámara web Logitech C920, que está conectada a una Odroid vía puerto usb.

Con la finalidad de proteger la integridad de los motores, las velocidades angulares de cada rueda se limitan con una función de saturación como se muestra a continuación:

$$f(\omega R, L_{ref}) = \omega_{R,L_{Acond}} = \begin{cases} 8 & \text{si } 60 < \omega R, L_{ref} \\ \frac{11 \omega R, L_{ref}}{120} + \frac{5}{2} & \text{si } \omega R, L_{ref} \in [\epsilon, 60] \\ 0 & \text{si } \omega R, L_{ref} \in [0, \epsilon] \\ -8 & \text{si } \omega R, L_{ref} < -60 \\ \frac{11 \omega R, L_{ref}}{120} - \frac{5}{2} & \text{si } \omega R, L_{ref} \in [-60, -\epsilon] \\ 0 & \text{si } \omega R, L_{ref} \in (-\epsilon, 0] \end{cases}, \quad (15)$$

donde: $\omega R, L_{ref}$ son las velocidades de ruedas obtenidas del control en regulación, $\omega_{R,L_{Acond}}$ son las velocidades de ruedas resultantes del proceso de acondicionamiento y $\epsilon = 0,01 \frac{rad}{s}$

En la Figura 7 se muestra la representación gráfica de la función de acondicionamiento de velocidades expresada en la ecuación (15).

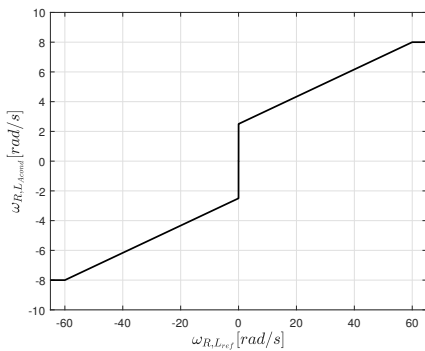


Figura 7: Acondicionamiento de la velocidad de las ruedas del robot móvil.

Además, para evitar colisiones, el control fue adaptado para detenerse cuando la cámara está a 10 cm del marcador,

$$e_x = \begin{cases} d & \text{si } d \leq 10cm \\ 0 & \text{si } d > 10cm \end{cases} \quad (16)$$

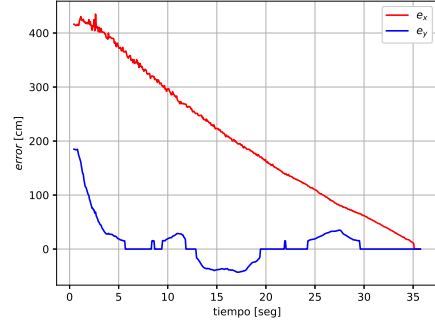


Figura 8: Errores locales del robot al colocarse tres metros atrás del ArUco.

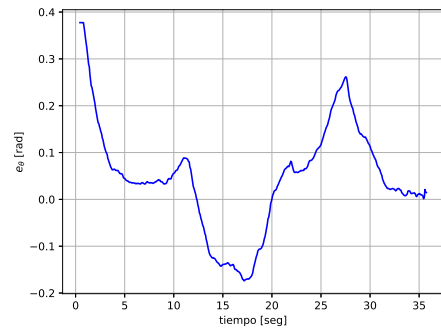


Figura 9: Comportamiento de la orientación del robot, medida con respecto al ArUco localizado por la cámara.

En la Figura 8 se observa el efecto de (16), evitando así que el marcador salga de la zona de visión o que el robot colisione con éste.

En la Figura 9 es notorio que el error e_θ se encuentra acotado entre $-\frac{\pi}{6}$ y $\frac{\pi}{6}$, esto por la naturaleza del campo de visión de la cámara; además se observa la relación entre e_θ y e_y (Figura 8)x

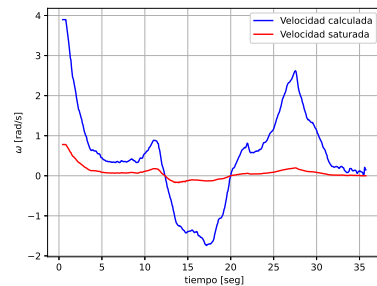


Figura 10: Velocidad angular exigida por el control contrastada con la velocidad angular real del robot durante la ejecución del experimento.

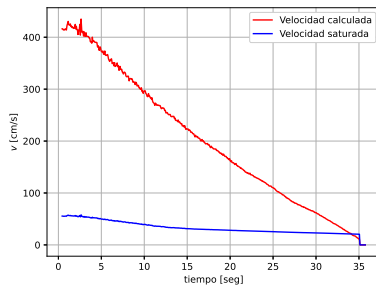


Figura 11: Velocidad lineal exigida por el control comparada con la velocidad real del robot durante la ejecución del experimento.

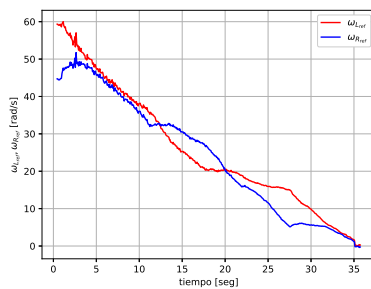


Figura 12: Señales de referencia hacia las ruedas del robot móvil.

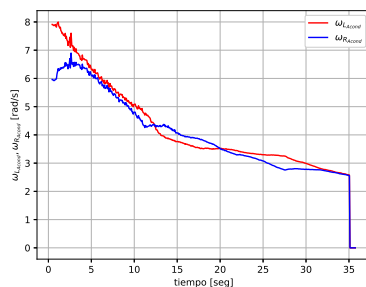


Figura 13: Acondicionamiento de las señales hacia las ruedas del robot móvil.

En la Figura 12 se muestran las velocidades angulares de la llantas del robot móvil requeridas por el control en regulación; sin embargo, tanto la magnitud de $\omega_{L_{ref}}$ como la magnitud de $\omega_{R_{ref}}$ no son alcanzables por los motores. Por lo tanto como se observa en la Figura 13 el acondicionamiento de señal mantiene las velocidades de llanta izquierda y llanta derecha dentro del rango seguro de operación de los actuadores.

5. Conclusiones

En este artículo se demostró que la ley de control propuesta junto a la estimación de la pose local con respecto a un marcador ArUco ofrece una estrategia de navegación funcional en un ambiente estructurado sin obstáculos para robots autónomos.

Además como se puede observar en el apartado de resultados, específicamente en 8 y 9 se comprobó que el uso de marcadores tipo ArUco como referencia es suficiente para llevar a cabo la tarea de regulación a un punto propuesta en la sección de control de este artículo.

El uso de marcadores en conjunto con sensores de visión comerciales, tales como cámaras web de alta definición, permiten una alternativa económica para la navegación de robots móviles en interiores, que tradicionalmente usan sensores más sofisticados como Lidar, sensores láser, entre otros.

Como parte importante de la estrategia de navegación se requiere considerar que para que el control realice su función es necesario que tenga un objetivo dentro del campo de visión de la cámara, en caso contrario, el robot debe entrar en una rutina de escaneo en 360 grados hasta encontrar un nuevo marcador.

Si bien, el uso de marcadores de referencia fue suficiente para realizar la navegación, quedan problemas como la oclusión y la limitante del campo de visión para realizarla; se propone el uso de sensores RGB-D con técnicas de visión por computadora como detección de puntos claves como trabajo futuro para compensar las deficiencias mencionadas.

Agradecimientos

Agradezco al CONACYT por su programa nacional de becas, gracias por el apoyo durante el postgrado. Gracias al CINESTAV-Saltillo por la formación en el grupo de Robótica y Manufactura Avanzada. Gracias a los doctores por su asesoría y ayuda incondicional.

Referencias

- Arduino (2022). Uno r3: Arduino documentation.
- Babinec, A., Jurišica, L., Hubinský, P., y Duchoň, F. (2014). Visual localization of mobile robot using artificial markers. *Procedia Engineering*, 96:1–9.
- De Luca, A., Oriolo, G., y Vendittelli, M. (2001). *Control of Wheeled Mobile Robots: An Experimental Overview*. Springer Berlin Heidelberg, Berlin, Heidelberg.
- de Oliveira Junior, A., Piardi, L., Bertogna, E. G., y Leitao, P. (2021). Improving the mobile robots indoor localization system by combining slam with fiducial markers. *2021 Latin American Robotics Symposium (LARS), 2021 Brazilian Symposium on Robotics (SBR), and 2021 Workshop on Robotics in Education (WRE)*.
- Guo, J., Wu, P., y Wang, W. (2020). A precision pose measurement technique based on multi-cooperative logo. *Journal of Physics: Conference Series*, 1607(1):012047.
- Gutiérrez, H., Morales-Díaz, A., y Nijmeijer, H. (2017). Synchronization control for a swarm of unicycle robots: analysis of different controller topologies: Synchronization control for a swarm of unicycle robots. *Asian Journal of Control*, 19.
- Khalil, H. K. (2015). *Nonlinear control / Hassan K. Khalil*. Pearson.
- Odroid (2022). Odroid-c4.
- OpenCV (2022). Detection of aruco markers.
- Poroykov, A., Kalugin, P., Shitov, S., y Lapitskaya, I. (2020). Modeling aruco markers images for accuracy analysis of their 3d pose estimation. *Proceedings of the 30th International Conference on Computer Graphics and Machine Vision (GraphiCon 2020). Part 2*.
- Zhang, C., Tao, D., Wang, L., y Wu, Y. (2021). Robot visual servo control system based on deep detection network and spatial pose estimation. *2021 33rd Chinese Control and Decision Conference (CCDC)*.