



Código	FPI-002
Objeto	Protocolo de presentación de proyectos de investigación SIGEVA UNLaM
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	4
Vigencia	12/11/2021

Unidad Ejecutora:
Departamento de Ingeniería e Investigaciones Tecnológicas

Programa de acreditación:

PROINCE

Título del Programa de Investigación¹:

Director del Programa:

Título del proyecto de investigación:

Navegación visual autónoma con celular

Director del proyecto:

Mg. Ing. Eterovic, Jorge

Co-Director del proyecto:

Integrantes del equipo:

Ing. Russo, Diego

Ing. Rzepa, José Antonio

Mg. Ing. Silvestri, Alejandro

Ing. Vaca, Gonzalo Nahuel (Graduado UNLaM)

Alumnos:

Bustamante, Juan Andrés

De Stefano, Matías

Pernas, Baltazar Agustín

Santos Ferreiro, Leandro Manuel

Sinopoli Alesio, Esteban Abel

Fecha de inicio:

1/1/2022

Fecha de finalización:

31/12/2023

¹ Completar solo en caso de que el presente proyecto se encuadre en el marco de un Programa de Investigación

1-Cuadro resumen de horas semanales dedicadas al proyecto por parte de director e integrantes del equipo de investigación:²

Rol del integrante	Nombre y Apellido	Cantidad de horas semanales dedicadas al proyecto
Director	Jorge Eterovic	4
Co-director		
Director de Programa		
Docente-investigador UNLaM	José Antonio Rzepa	4
Docente-investigador UNLaM	Diego Russo	4
Docente-investigador UNLaM	Alejandro Silvestri	4
Investigador externo ³		
Asesor-Especialista externo ⁴		
Graduado de la UNLaM ⁵	Gonzalo Nahuel Vaca	4
Estudiante de carreras de posgrado (UNLaM) ⁶		
Alumno de carreras de grado (UNLaM) ⁷	Juan Andrés Bustamante	4
Alumno de carreras de grado (UNLaM)	Matías De Stefano	4
Alumno de carreras de grado (UNLaM)	Baltazar Agustín Pernas	4
Alumno de carreras de grado (UNLaM)	Leandro Manuel Santos Ferreiro	4
Alumno de carreras de grado (UNLaM)	Esteban Abel Sinopoli Alesio	4
Personal de apoyo técnico administrativo		

2 Incluir todos los integrantes del equipo de investigación, agregando tantas filas para cada rol de integrante del equipo de investigación como sea necesario.

3 Deberá adjuntar FPI 28, 29 y 30 debidamente firmados.

4 Idem nota 2.

5 Idem nota 2

6 Adjuntar certificado de materias aprobadas de estudiantes de carrera de posgrado.

7 Adjuntar certificado de materias aprobadas de estudiantes de carrera de grado.

2-Plan de investigación

2. Tipo de actividad I+D: Desarrollo experimental

2.1. Resumen del Proyecto:

Los AMR, Robots Móviles Autónomos, son vehículos autónomos para depósitos, instalaciones industriales y otros ámbitos privados. Los AMR comerciales utilizan un sensor LIDAR y un algoritmo de SLAM para la autolocalización. Siendo el LIDAR un sensor relativamente caro, en laboratorios se está experimentando intensamente con una alternativa de mucho menor costo, la tecnología Visual SLAM, que sustituye el LIDAR por una webcam.

Los sistemas de Visual SLAM tienen alta demanda computacional que impide su ejecución en celulares, y el AMR debe cargar una PC. Este proyecto divide la carga computacional de manera que un celular montado en el AMR capte y procese la imagen (lo que se conoce como *edge computing*) para enviar los datos por WIFI a una PC de escritorio que se ocupe del resto.

Complementariamente en el marco de este proyecto se construirá un birrueda eléctrico y se implementará su sistema de control. El birrueda propuesto es un vehículo de muy bajo costo y amplio espacio para experimentación en robótica. El conjunto de un birrueda eléctrico con un sistema de autoposicionamiento basado en Visual SLAM constituye una plataforma de bajo costo para probar soluciones de AMR, muy apto para la experimentación en grado, para la investigación, e incluso para sugerir su replicación en colegios técnicos. Podría ser también la base de prototipado de productos comerciales y soluciones reales a medida.

2.2. Palabras clave: Visual-SLAM, ORB-SLAM2, birrueda

2.3 Resumen del Proyecto (inglés):

AMR, Autonomous Mobile Robots, are autonomous vehicles for warehouses, industrial facilities and other private areas. Commercial AMRs use a LIDAR sensor and a SLAM algorithm for autolocation. Since LIDAR is a relatively expensive sensor, laboratories are experimenting intensively with a much lower cost alternative, Visual SLAM technology, which replaces LIDAR with a webcam.

Visual SLAM systems have a high computational demand that prevents their execution on cell phones, and the AMR must carry a PC. This project divides the computational load so that a cell phone mounted on the AMR captures and processes the image (what is known as edge computing) to send the data via WIFI to a desktop PC that takes care of the rest.

In addition, within the framework of this project, an electric diwheel will be built and its control system will be implemented. The proposed diwheel is a very low cost vehicle with ample space for robotics experimentation. The set of an electric diwheel with an auto-positioning system based on Visual SLAM constitutes a low-cost platform to test AMR solutions, very suitable for experimentation in engineering education, for research, and even to suggest its replication in technical colleges. It could also be the basis for prototyping commercial products and real customized solutions.

2.4 Palabras clave (inglés): Visual-SLAM, ORB-SLAM2, diwheel

2.5 Disciplina desagregada:

2107 INGENIERIA-ELECTRONICA

2.6 Campo de aplicación: 8 Industrial

2.7 Especialidad: AMR, vehículos autónomos para predios privados

2.8 Estado actual del conocimiento:

La tecnología Visual SLAM alcanzó su madurez en 2015 con los papers ORB-SLAM y LSD-SLAM. Desde entonces se ha venido desarrollado intensamente hacia su aplicación práctica, constituyendo un campo de investigación activa cuyos publicaciones referentes son ORB-SLAM2, OpenVSLAM y ORB-SLAM3, esta última de 2021.

La demanda computacional de estos sistemas los hacen inviables para su uso práctico en celulares. Según lo investigado éste es el primer proyecto que apunta a dividir la demanda computacional encargando al celular lo que se conoce como edge computing, y delegando a una PC de escritorio por WIFI la computación pesada.

En cuanto al vehículo birrueda, Alejandro Silvestri ha dirigido el diseño y desarrollo de cinco versiones de birrueda en facultades de ingeniería.

Sobre Visual SLAM:

SLAM significa Mapeo y localización simultáneos; es el problema de combinar sucesivas lecturas de un sensor en un móvil (por ejemplo un AMR) para reconocer y mapear los alrededores y ubicarse en ese mapa. Visual SLAM es el SLAM con una cámara como sensor. En 2007 dos papers significaron el punto de partida de Visual SLAM: PTAM y monoSLAM. Ambos presentaron los primeros sistemas de visual SLAM que corrían en tiempo real (sobre una computadora muy potente).

En los años siguientes sistemas de Visual SLAM mejoraron la precisión al tiempo que reducían la demanda computacional. Afloraron diversos métodos, siendo los más destacados los denominados "directo" e "indirecto". Entre 2014 y 2015 Visual SLAM alcanzó la madurez en las publicaciones ORB-SLAM y LSD-SLAM. Ambos trabajos fueron el máximo exponente de cada una de los dos métodos dominantes, y ambos lograron minimizar la demanda computacional pudiendo ejecutarse en una PC común sin GPU, a la vez que alcanzaron máxima precisión de mapeo ganando varias competencias.

Desde entonces la comunidad científica se volcó en busca de nuevos objetivos: incorporar nuevos tipos de cámaras (gran angular, omnidireccional, cámara de profundidad), nuevos sensores (acelerómetros 6D) y aumentar las facilidades acercando estos desarrollos al uso práctico. Mientras que el método directo dio lugar a excelentes sistemas de odometría visual, el método indirecto domina la escena de Visual SLAM en 2022.

Sobre edge computing:

En 2022 la tecnología web se encuentra madura para implementar edge computing en un navegador del celular.

El celular debe captar la imagen de la cámara y manipularla de manera eficiente, lo que requiere el uso de webassembly.

2.9. Problemática a investigar:

Este proyecto propone el diseño y construcción de un prototipo de vehículo autónomo que sirva como base de pruebas de proyectos académicos y productos comerciales. El desarrollo de este vehículo hace hincapié en la facilidad de implementación y en el bajo costo de replicación. El prototipo se descompone en tres partes:

- Parte mecánica:
 - diseño de un vehículo eléctrico birrueda: vehículo diferencial, con dos ruedas con motorización independiente
 - requiere una transmisión adecuada (reducción de velocidad del motor a la rueda)
 - emplea una gran cantidad de partes plásticas a medida impresas en 3D
- Parte electrónica
 - sistema de potencia, control de los motores de continua con puente H y PWM
 - sensado de velocidad de cada rueda
 - sistema de control con microcontrolador

- Parte informática
 - implementación de un sistema de autolocalización por Visual SLAM dividido en “edge” (captura y procesamiento de imagen en celular) y “backend” (procesamiento pesado del resto de Visual SLAM)
 - basado en algún sistema de Visual SLAM existente, de código abierto, representando el estado del arte

El vehículo birrueda es notorio por lo disruptivo de su diseño. Su aparente simpleza posibilita una construcción de bajo costo pero esconde una dinámica compleja que requiere análisis profundo y dificulta su control. El diseño propuesto aborda el problema con un sistema de control lineal, dejando como futuras mejoras posibles el diseño de un control no lineal a medida.

La autolocalización es la capacidad de un vehículo autónomo de saber dónde se encuentra, en qué coordenadas de un mapa se halla. El sistema autolocalización es uno de los aspectos más innovadores de este proyecto, por su particular formulación dividida entre un celular y una PC, en la que un celular montado en el birrueda capta la imagen del entorno, y el procesamiento pesado de Visual SLAM se realiza en una PC de escritorio. Todo este proyecto gira alrededor de este punto particular, que facilitará la autolocalización visual para cualquier vehículo, incluso aquellos pequeños muy usados en prácticas de laboratorio.

2.10. Objetivos:⁸

Objetivos generales:

Producir un birrueda de bajo costo con capacidad de autolocalización, como plataforma de código abierto que sirva al desarrollo continuo, tanto para su uso académico como para la experimentación y para el prototipado de productos.

Este prototipo se puede controlar de forma remota e inalámbrica, demostrando su capacidad de autolocalización en un ambiente previamente mapeado. Este birrueda constituye una plataforma para experimentar soluciones de AMR.

El componente que no se desarrolla en este proyecto es el sistema de navegación. Este sistema se ocupa de utilizar la localización sensada para conducir al birrueda a lo largo de una trayectoria. Para realizar pruebas se proporcionará un sistema de navegación elemental, muy mejorable. El posterior desarrollo de un sistema de navegación avanzado amerita un proyecto propio, se considera uno de los proyectos a realizar sobre la plataforma resultante del presente, y puede ser el objetivo de un proyecto de grado.

Objetivos particulares:

Dividir el sistema de visual SLAM en dos partes (edge y backend), de modo que el sensor montado en el vehículo se reduzca a un simple celular, mientras que la inteligencia y poder de cómputo se delega a una PC de escritorio. Este objetivo es la esencia del proyecto.

Complementariamente, se procura **diseñar un birrueda eléctrico de bajo costo** que sirva de vehículo experimental.

Finalmente, se requiere **diseñar un sistema de control que ejecute órdenes básicas de navegación**, como avanzar en línea recta a determinada velocidad, o recorrer una determinada distancia con un cierto radio de giro. Se trata de un sistema de control MIMO, con dos entradas desde sendos encoders y dos salidas hacia los motores.

2.11. Marco teórico:

⁸ Detallar objetivo general y objetivos específicos.

Hay dos grandes grupos de vehículos autónomos terrestres: automóviles que viajan en rutas públicas, muchas veces desconocidas (no visitadas previamente); y vehículos de depósitos e instalaciones privadas, denominado AMR. Este último tipo es el de interés en este proyecto: suelen ser vehículos pequeños que conviven con tránsito peatonal y se mueven en ambientes privados, sin demarcaciones, y previamente relevados.

El GPS no brinda la precisión requerida por un vehículo para mantenerse en su carril. Los AMR observan sus alrededores con sensores para reconocerlos y autolocalizarse (descubrir su localización en un mapa conocido). SLAM (Simultaneous Localization And Mapping) es el nombre del problema general de deducir el movimiento propio (ego-motion) a partir de sucesivas lecturas del sensor. Es un problema geométrico sin solución analítica que se debe calcular en tiempo real y demanda mucha computación.

Visual SLAM es la rama de SLAM que emplea una cámara como sensor: identifica elementos visuales y los triangula a partir de imágenes obtenidas desde diferentes localizaciones de la cámara, generando un mapa 3D de los alrededores. Esta tecnología resuelve el problema de la autolocalización, con un sensor de bajo costo.

Los sistemas de Visual SLAM tienen alta demanda computacional y no han logrado ejecutarse satisfactoriamente en celulares.

2.12. Hipótesis de trabajo o los supuestos implícitos (según corresponda al diseño metodológico) :⁹

Este proyecto plantea el diseño e implementación de prototipos como solución al problema planteado.

Solución planteada para la división del proceso de visual SLAM:

En un sistema de Visual SLAM la imagen capturada por la cámara ingresa en un proceso de visión artificial que extrae las *características* usadas para el mapeo y la localización. Apenas se extraen estas características, la imagen se descarta.

Se propone dividir este proceso delegando la extracción de tales características al dispositivo móvil montado en el vehículo. El celular envía luego por WIFI estas características a la PC de escritorio, que realizará la parte pesada del mapeo y localización, enviando como respuesta la pose del vehículo en el mapa. Esta respuesta se puede enviar al celular, o directamente al sistema de control del vehículo.

Éste es el proceso que debe ejecutar el celular:

1. Captar la imagen con la cámara.
2. Convertirla a blanco y negro.
3. Detectar keypoints con el detector de características FAST.
 - a. *Faster and better: a machine learning approach to corner detection, Rosten 2010*
4. Repetir la detección con parámetros más permisivos, en cuadrantes donde se hayan detectado pocos keypoints.
5. Filtrar keypoints amontonados pues resultan redundantes, de manera de alivianar el trabajo al resto del sistema Visual SLAM.
6. Extraer descriptores BRIEF orientados con ORB, solamente para los keypoints que superaron el filtro anterior.
 - a. *BRIEF: Binary Robust Independent Elementary Features, 2010*
 - b. *ORB: an efficient alternative to SIFT or SURF, 2012*

⁹ En proyectos de desarrollo tecnológico puede ser reemplazada una hipótesis de trabajo por la propuesta de solución al problema de investigación mediante el diseño de un prototipo o elemento equivalente.

7. Opcionalmente antidistorsionar (compensar distorsiones de lente) las coordenadas de los keypoints para compensar distorsiones de lente y obtener una colinealidad.
8. Opcionalmente clasificar los descriptores con Bag Of Words.
 - a. *Bag-of-Words-driven Single Camera Simultaneous Localisation and Mapping, 2010*

Los pasos imprescindibles son los que se realizan sobre la imagen. Los pasos posteriores son opcionales, y contribuyen a aliviar el cómputo en la PC.

Los obstáculos técnicos en el edge a superar son:

- el sistema edge en el celular se proporciona vía web; de esta manera es multiplataforma y se evita la instalación del software en el celular
- este proceso requiere alta velocidad, se hará en el navegador del celular usando webassembly
- el código de este proceso se escribirá en C++ y se compilará con emscripten para obtener código webassembly ejecutable en una página web
- se intentará usar la biblioteca OpenCV.js para operar sobre las imágenes. Se recompilará la biblioteca activando funciones particulares necesarias que no se encuentran en la versión pública.
- si no resulta posible conectar el código webassembly del pipeline con OpenCV.js, entonces se incorporará la biblioteca C++ OpenCV y se compilará de forma monolítica.

El código edge presentará una API javascript para que los desarrolladores puedan interactuar con el sistema de visual SLAM, alimentando imágenes y recibiendo la localización.

Los obstáculos técnicos del backend son:

- el pipeline descrito es conceptual, pero los sistemas de visual SLAM no lo tienen debidamente encapsulado; separarlo requiere “cirugía mayor” del código C++ de la mayoría de los sistemas disponibles, sea ORB-SLAM 1, 2 o 3, OpenVSLAM o cualquier otro
- la mayoría de estos sistemas se desarrollaron en entornos científicos, y carecen de herramientas prácticas elementales; al sistema se le deberá agregar una manera práctica de guardar en archivo mapas relevados para ser usados en siguientes oportunidades
- hace falta un método para trazar rutas manuales en los mapas relevados para un sistema de navegación elemental

Solución para el birrueda:

Un segundo objetivo es lograr un birrueda de bajo costo. El camino para hacerlo emplea ruedas de bicicleta, universalmente disponibles y mucho más baratas que la fabricación de ruedas ad hoc. Hay tres formas generales de transmisión mecánica del movimiento del motor a la rueda: vía piñón y cadena de bicicleta, por engranajes impresos en 3D, por fricción. Se evaluarán las tres procurando la solución efectiva de menor costo.

El birrueda consiste de dos motores eléctricos independientes (uno para cada rueda), sistema de reducción mecánica para acoplar el motor con la rueda, batería de 24V. El diseño mecánico debe contemplar el montaje de encoders en cada rueda, de un compartimiento para alojar la electrónica, y del soporte para un celular con cámara debidamente orientada para funcionar como sensor visual.

El diseño se debe acompañar del análisis cinemático y dinámico del sistema. El análisis dinámico se lleva a cabo sobre mediciones experimentales sobre el birrueda ya montado.

Solución para el sistema de control:

Un birrueda es un vehículo de control altamente inestable, y requiere un sistema de control electrónico para poder conducirlo. Este sistema de control es medianamente complejo, se plantea en dos niveles:

- Bajo nivel, control clásico tipo PI (proporcional e integral): un sistema de control independiente para cada rueda, que mide un encoder y gobierna la potencia entregada a un motor
- Nivel superior, MIMO (multiple inputs, multiple outputs): un sistema complejo cuyos setpoints son velocidad de avance y radio de giro, y sus salidas gobiernan los setpoints de los sistemas de bajo nivel

Este proyecto contempla el desarrollo e implementación de un sistema lineal, que será ajustado con los métodos clásicos de relevamiento de planta a través de la respuesta al escalón y a la rampa. Siendo el birrueda un sistema no lineal, el control no lineal a medida queda fuera del alcance del presente proyecto, como una oportunidad de mejora que amerita un proyecto propio usando el prototipo desarrollado en éste.

Futuros proyectos:

El resultado prometido por este proyecto es un prototipo apto para experimentación. En los párrafos anteriores se mencionaron dos posibles proyectos futuros usando este prototipo:

- sistema de navegación, un problema no trivial que se puede resolver de muchas maneras
- sistema de control no lineal, otro problema no trivial que requiere una solución teórica avanzada de nivel de posgrado

2.13. Metodología:

El proyecto es la combinación de varias partes desarrolladas a medida que requiere ajustes derivados de mediciones en pruebas, primero para cada una de las partes y luego para el conjunto. Esta metodología fue empleada numerosas veces en los últimos años en desarrollo de birruedas (no autónomos) en materias de grado.

Pruebas y mediciones sobre la división de Visual SLAM:

1. performance de edge computing (tiempo de procesamiento de imagen en celular)
2. performance de las comunicaciones (lag y tiempo total de transmisión de los datos extraídos de la imagen)
3. visualización de los keypoints suministrados por el celular, superpuestos a la imagen para comprobar la integridad de los datos transmitidos
4. testing del código de backend, suministrando un flujo de datos obtenido del celular
5. visualización del mapa creado y confirmación de que el autopoicionamiento sea correcto
6. medición del error promedio de autopoicionamiento contrastando medición con posición real

Pruebas y mediciones del birrueda:

El birrueda requiere un sistema de control básico, que puede ser de lazo abierto, para realizar las primeras pruebas, en paralelo al desarrollo del sistema de control final. Con el sistema de control básico se realizan estos ensayos:

1. ensayo de velocidad máxima de cada rueda en vacío (con rampa de arranque), procurando comprobar groseramente la velocidad esperada y midiendo con precisión la velocidad alcanzada

2. ensayo de recorrido circular (una rueda detenida) en el suelo, con diferentes potencias aplicadas al motor, procurando relevar la tabla que relaciona la modulación de ancho de pulso con la velocidad final
 - a. el recorrido circular tiene la ventaja de poder realizarse en un espacio pequeño, sin posibilidad de que el vehículo se dispare fuera de control
3. ensayo de recorrido recto (ambas ruedas con la misma potencia) en el suelo con un timer de apagado, procurando obtener la función de velocidad en el tiempo para varias potencias aplicadas
 - a. este ensayo requiere un espacio amplio y controlado (sin peatones ni obstáculos), el birrueda gira inesperadamente y sale en cualquier dirección
 - b. la falta de un sistema de control realimentado hace virtualmente imposible un recorrido recto
 - c. un timer de 5 a 10 segundos (en función del espacio disponible) apaga el vehículo para que no se escape
 - d. el birrueda debe disponer de un “kill switch”, atado a una cuerda que cuando se tensa apaga el vehículo

Ensayos del sistema de control:

1. test de línea recta, en la que el vehículo debe recorrer una distancia determinada; usualmente durante el arranque cambia de dirección
2. test de arranque recto, similar al anterior procurando que el vehículo no cambie su dirección inicial
3. test de radio de giro, haciendo un movimiento circular, en el que se mide la deriva (el desplazamiento no intencional del centro de giro)
4. test cuadrado, en la que el vehículo alterna un avance recto con un giro de 90°, describiendo un cuadrado con vértices redondeados; este test verifica el correcto control sobre distancias y ángulos de giro
5. test de velocidad constante, en una línea recta y en circuito circular, luego de alcanzar la velocidad de régimen, se mide su estabilidad: las deficiencias en el control producen oscilaciones lentas en la velocidad del vehículo

2.14. Bibliografía:

Este proyecto se basa exclusivamente en papers que han publicado el código para pruebas como código abierto. A continuación se citan los papers principales de Visual SLAM, y un paper con el análisis cinemático y dinámico del birrueda (diwheel en inglés):

Mur Artal R et al 2015, “ORB-SLAM: A Versatile and Accurate Monocular SLAM System”, IEEE Transactions on Robotics, Volume 31 issue 5, accedido en Marzo de 2016, <https://ieeexplore.ieee.org/document/7219438>

Mur Artal R et al 2017, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras”, IEEE Transactions on Robotics Volume: 33 Issue: 5, accedido en Marzo de 2018, <https://ieeexplore.ieee.org/document/7946260>

Shinya Sumikura et al, 2019, “OpenVSLAM: A Versatile Visual SLAM Framework”, Proceedings of the 27th ACM International Conference on Multimedia, accedido en Agosto de 2019, <https://arxiv.org/abs/1910.01122>

Campos C et al, 2021, “ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM” arxiv preprint, accedido en Diciembre de 2021, <https://arxiv.org/abs/2007.11898>

Cazzolato B et al 2009, "Control of an electric diwheel", accedido en Febrero de 2020, https://www.researchgate.net/publication/228909424_Control_of_an_electric_diwheel

Estos enlaces llevan a documentos propios que reúnen referencias a más de un centenar de papers complementarios:

- [Detectores y descriptores](#)
- [Loop closure & place recognition](#)
- [Homografía](#)
- [Odometría visual](#)
- [Pose y Perspective from N points](#)
- [Solvers](#)
- [Structure from motion](#)
- [Triangulación y geometría epipolar](#)
- [Visual SLAM](#)

2.15. Programación de actividades (Gantt):¹⁰

Actividad	Responsable	1º semestre	2º semestre	3º semestre	4º semestre
Diseño y construcción del birrueda	Diego Russo	X	X	X	
Mediciones y ajuste del diseño (rediseño y reimplementación)	Diego Russo			X	X
Diseño e implementación del sistema de control	Antonio Rzepa	X	X	X	
Mediciones y ajuste del sistema de control	Antonio Rzepa			X	X
Diseño, programación e implementación de la electrónica y sistema embebido	Gonzalo Vaca	X	X		
Reprogramación del sistema embebido para nuevo sistema de control, comunicaciones e interconexión con celular	Gonzalo Vaca		X	X	X
División de Visual SLAM	Silvestri, Alejandro	X	X	X	
Test del sistema dividido y ajuste	Silvestri, Alejandro		X	X	X
Pruebas sobre Birrueda	Silvestri, Alejandro			X	X
Ajuste del sistema dividido a partir de pruebas sobre birrueda	Silvestri, Alejandro				X

¹⁰ Definir la programación de actividades para cada objetivo específico, y las personas responsables de su ejecución.

Desarrollo de herramientas accesorias	Silvestri, Alejandro			X	X
---------------------------------------	----------------------	--	--	---	---

2.16. Resultados en cuanto a la producción de conocimiento:

Este proyecto pondrá a prueba dos conceptos novedosos:

- la viabilidad y conveniencia de dividir Visual SLAM en edge computing y backend, con tablas de performance de cada parte
- la viabilidad del uso de birruedas como manera de abaratar el vehículo a costa de complicar el sistema de control

De manera integral, demostrará la viabilidad práctica de un AMR de bajo costo.

2.17. Resultados en cuanto a la formación de recursos humanos:

El equipo está integrado por docentes que pertenecen a cátedras de la carrera de Ingeniería en Informática e Ingeniería Electrónica de la UNLaM que se están formando como investigadores. Como parte del proyecto se organizarán actividades prácticas de capacitación en la metodología de investigación aplicada. Se incorporaron cinco alumnos de grado de la carrera de Ingeniería en Informática.

2.18. Resultados en cuanto a la difusión de resultados:

Se escribirán trabajos para difundir los resultados en diferentes congresos:

- WICC – Workshop de Investigadores en Ciencia de la Computación.
- CoNalISI – Congreso Nacional de Ingeniería Informática y en Sistemas de Información.
- CACIC – Congreso Argentino de Ciencias Informáticas y Computación.
- Publicaciones Periódicas de robótica.

2.19. Resultados en cuanto a transferencia hacia las actividades de docencia y extensión:

El prototipo esperado permitiría ser replicado en colegios técnicos (como práctica en sí misma y como prototipo para pruebas), se usará en materias de ingeniería electrónica y con grandes posibilidades de ser usado también en informática para proyectos de software, e incluso en ingeniería industrial para simulación de AMR en depósitos.

El prototipo pretende ser la base para futuros proyectos, incluyendo el prototipado de productos reales y soluciones para empresas. Servirá además para mostrar y hacer difusión de varias carreras de ingeniería.

2.20. Resultados en cuanto a la transferencia de resultados a organismos externos a la UNLaM: No aplica.

2.21. Vinculación del proyecto con otros grupos de investigación del país y del exterior:

Se espera vincular el proyecto con grupos de investigación de otras Universidades de Gestión Pública y Privada interesados en esta temática.

2.22. Destinatarios:

Tipo de destinatario		Subtipo de destinatario ¹¹	¿Cuál? Especificar	Demandante ¹²	Adoptante ¹³
Sector Gubernamental	Gobiernos	Del Poder Ejecutivo nacional			
		Del Poder Ejecutivo provincial			
		Del Poder Ejecutivo municipal			
	Otras Instituciones gubernamentales	Poder Legislativo en sus distintas jurisdicciones			
Poder Judicial en sus distintas jurisdicciones					
Sector Salud		Hospitales, centros comunitarios de salud y otras entidades del sistema de atención			
Sector Educativo	Sistema universitario		UNLaM		X
	Sistema de educación básica y secundaria		EEST N° 6 Chacabuco	X	X
	Sistema de educación terciaria				
Sector Productivo	Empresas				
	Cooperativas de trabajo y producción				
	Asociaciones del Sector				
Sociedad Civil	ONG's y otras organizaciones sin fines de lucro				
	Comunidades locales y particulares				

3-Recursos Existentes¹⁴

11 Marcar con una X

12 Demandante: entidad administrativa de gobierno nacional, provincial o municipal constituida como demandante externo de las tecnologías desarrolladas, que determina la necesidad del proyecto por su importancia social. Marcar con una X

13 Adoptante: beneficiario o usuario en capacidad de aplicar los resultados desarrollados (organismos gubernamentales de ciencia y tecnología nacionales o provinciales; universidades e institutos universitarios de gestión pública o privada; empresas públicas o privadas; entidades administrativas de gobierno nacionales, provinciales o municipales; entidades sin fines de lucro; hospitales públicos o privados; instituciones educativas no universitarias; y organismos multilaterales. Marcar con una X

14 Antes de confeccionar el presupuesto del proyecto, será necesario que el Director incluya en esta tabla si dispone de recursos adquiridos con fondos de proyectos anteriores (equipamiento, bibliografía, bienes de consumo, etc.) a ser utilizados en el proyecto a presentar, y además se recomienda consultar en la Unidad académica la disponibilidad de recursos existentes factibles de ser utilizados en el presente proyecto.

Descripción/ concepto	Cantidad	Observaciones
PC de escritorio o notebook	1	Se requiere una PC, que puede ser una notebook de los integrantes del proyecto
Celular	1	Los integrantes del proyecto utilizarán sus propios celulares en las pruebas
Impresión 3D	1	Las piezas mecánicas diseñadas en este proyecto se imprimirán en impresoras del DIIT

4-Recursos financieros¹⁵

	Rubro	Año 1	Año 2	Total
Gastos de capital (equipamiento)	a) Equipamiento (1)	\$75.000	\$75.000	\$150.000
	b) Licencias (2)			
	c) Bibliografía (3)			
	Total Gastos de Capital	\$75.000	\$75.000	\$150.000
Gastos corrientes (funcionamiento)	d) Bienes de consumo			
	e) Viajes y viáticos (4)			
	f) Difusión y/o protección de resultados (5)			
	g) Servicios de terceros (6)			
	h) Otros gastos (7)			
	Total Gastos Corrientes	\$ 0,00	\$ 0,00	\$ 0,00
	Total Gastos (Capital + Corrientes)	\$75.000	\$75.000	\$150.000

Precios testigo

Este listado es de referencia, no es exhaustivo, el modelo exacto de cada parte se define durante el diseño del prototipo, una tarea que es parte del proyecto. El listado no incluye electrónica y materiales de bajo costo, como componentes, caños, tornillos, etc.

- Llantas x2
 - \$29.800 (\$14.900 c/u)
 - https://articulo.mercadolibre.com.ar/MLA-1127768683-rueda-rodado-29-mti-shimano-7v-_JM
- Cubiertas x2
 - \$6.000
 - https://articulo.mercadolibre.com.ar/MLA-920444626-kit-2-cubiertas-bicicletas-camaras-rodado-26-190-_JM
- Motorreductor x2
 - \$31.000 (\$15.500 c/u)
 - https://articulo.mercadolibre.com.ar/MLA-863464931-motorreductor-motor-reductor-transmision-a-90-12v-24v-_JM
- Cámara HD
 - \$47.000
 - https://articulo.mercadolibre.com.ar/MLA-614010339-camara-web-logitech-hd-c525-c-microfono-hd-720p-_JM
- Filamento PLA x5Kg
 - \$74.200 (\$14.840 x 1 Kg)

15 Justificar presupuesto detallado. Para compras de un importe superior a \$15000.- se requieren tres presupuestos. (Resolución Rectoral N°177/2021.)

- https://articulo.mercadolibre.com.ar/MLA-843961287-filamento-pla-175mm-1kg-color-foto-1-imp-3d-0e5xnuvy-_JM
- Microcontrolador
 - \$14.800
 - https://articulo.mercadolibre.com.ar/MLA-1119820309-visualizacion-oled-wifi-kit32-arduino-big-venta-esp32-_JM
- Puente H x4
 - \$38.400 (\$9.600 c/u)
 - https://articulo.mercadolibre.com.ar/MLA-905136022-10-x-doble-puente-h-driver-l298n-motor-dc-arduino-_JM
- Baterías x2
 - \$23.600 (\$11.800 c/u)
 - https://articulo.mercadolibre.com.ar/MLA-875525297-bateria-moto-gel-ytx12-bs-bosch-btx12-12v-10ah-vzh-_JM
- Cargador baterías
 - \$11.800
 - https://articulo.mercadolibre.com.ar/MLA-624869218-bateria-para-motos-yuasa-yb7c-a-12v-8ah-vzh-srl-_JM

Aclaraciones sobre rubros del presupuesto

- 1 Equipamiento: Equipamiento, repuestos o accesorios de equipos, etc.
- 2 Licencias: Adquisición de licencias de tecnología (software, o cualquier otro insumo que implique un contrato de licencia con el proveedor).
- 3 Bibliografía: En el caso de compra de bibliografía, ésta no debe estar accesible como suscripción en la Biblioteca Electrónica.
- 4 Viajes y viáticos: Viajes y viáticos en el país: Gastos de viajes, viáticos de campaña y pasantías en otros centros de investigación estrictamente listados en el proyecto. Gastos de viaje en el exterior: (no deberán superar el 20% del monto del proyecto).
- 5 Difusión y/o protección de resultados: Ej.: (Gastos para publicación de artículos, edición de libros inscripción a congresos y/o reuniones científicas).
- 6 Servicios de terceros: Servicios de terceros no personales (reparaciones, análisis, fotografía, etc.).
- 7 Otros gastos: Incluir, si es necesario, gastos a realizar que no fueron incluidos en los otros rubros.

4.1 Origen de los fondos solicitados

Institución	% Financiamiento
UNLaM	100%



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Departamento:
Ingeniería e Investigaciones Tecnológicas

Programa de acreditación:
CyTMA2

Código del Proyecto:
C2-ING-095

Título del proyecto:
Navegación visual autónoma con celular

Director:
Mag. Ing. Jorge Esteban Eterovic

Codirector:
Mag. Ing. Jorge Alejandro Silvestri

Integrantes:
Ing. José Antonio Rzepa
Ing. Diego Russo
Ing. Gonzalo Vaca

Alumnos de grado:
Bustamante, Juan Andrés
De Stefano, Matías
Pernas, Baltazar Agustín
Santos Ferreiro, Leandro Manuel
Sinopoli Alesio, Esteban Abel

Resolución Rectoral de acreditación: N° 616/22

Fecha de inicio: 01/01/2022



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Fecha de finalización: 31/12/2023



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

A. Desarrollo del proyecto

A.1. Grado de ejecución de los objetivos inicialmente planteados, modificaciones o ampliaciones u obstáculos encontrados para su realización (desarrolle en no más de dos (2) páginas)

El sistema de posicionamiento visual con celular se ha desarrollado por completo, y se han elaborado tres publicaciones.

El sistema consta de tres módulos cuyos repositorios se encuentran alojados en <https://github.com/orgs/Birrueda/repositories> :

- Backend
- Web processor
- WASM processor

Los README de los repositorios se escribieron en inglés, para facilitar su publicación internacional.

El hardware asociado, un vehículo birrueda, fue desarrollado en un 100%, faltando solamente el software de su microcontrolador. Se ha cedido al laboratorio de electrónica.



La demora en la disponibilidad de fondos, que ocurrió a fines de 2022, redujo en un año el tiempo para el desarrollo del hardware y sus pruebas. Se pudo diseñar y construir, pero no se llegó a programar y testear.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

A continuación se resume el corazón del proyecto, denominado Visual SLAM híbrido:

Un robot móvil autónomo (AMR: Autonomous Mobile Robot) es un vehículo autónomo para predios, interiores y otros lugares controlados. A diferencia de los automóviles autónomos (self-driving cars), los AMR no cuentan con un camino claramente visible; tampoco cuentan con marcas de ruta ni con la asistencia de balizas. En su lugar tienen en memoria un mapa y la trayectoria que deben seguir, y emplean un sensor para una localización precisa. Los AMR comerciales adoptan costosos sensores LiDAR cuya salida es procesada por sistemas SLAM (Simultaneous Localization And Mapping) computacionalmente intensivos que les permiten deducir su localización a partir de un relevamiento de los alrededores del robot. Tanto el LiDAR como la computadora que procesa SLAM se encuentran a bordo del robot.

Una alternativa a la dupla LiDAR y SLAM son los sistemas de SLAM visual o vSLAM, que emplean una cámara de video como sensor. La tecnología vSLAM sigue siendo experimental pero está alcanzando un alto grado de madurez. Estos sistemas evitan el elevado costo de un LiDAR, reemplazándolo por una cámara de bajo costo. Los sistemas vSLAM son computacionalmente intensivos y no pueden ejecutarse en un celular, por lo que un AMR con vSLAM requiere una computadora a bordo.

Con la intención de evitar montar una computadora a bordo, se ha estudiado la posibilidad de montar una cámara WIFI (que se puede implementar con un celular), pero presenta estos serios inconvenientes:

- El flujo de video comprimido tiene un retardo significativo, superior a un segundo, y como la compresión tiene pérdidas la calidad de la imagen se degrada, reduciendo la capacidad de localización.
- Un flujo de imágenes sin comprimir ocupa demasiado ancho de banda, usualmente no disponible en las redes WIFI.

Este trabajo de investigación propone un sistema de vSLAM partido, con un celular a bordo que capture las imágenes y realice un preprocesamiento, extrayendo el denominado descriptor de imagen, para enviarlo por WIFI a una computadora estática, fuera del robot, que ejecuta la parte restante del sistema vSLAM y obtiene la localización del robot. El descriptor de imagen tiene un tamaño en bytes significativamente menor que el de la propia imagen, y las pruebas confirman que se pueden enviar por WIFI sin inconvenientes.

Este sistema híbrido adopta un celular como dispositivo de borde (edge computing), fácil de montar de una variedad de robots, habilitando la capacidad de localización visual para robots pequeños y de prototipado, como las aspiradoras autónomas y los robots Arduino, y permitiendo ejecutar vSLAM en una computadora de escritorio o incluso en la nube.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

La división de tareas elegida habilita el uso de celulares de baja gama, lo que facilita la adopción de esta tecnología. Las pruebas con celulares de muy baja gama logran procesar 5 imágenes por segundo, suficiente para la conducción autónoma de los AMR.

Para mayor compatibilidad el procesamiento en celular se realiza enteramente en una página web en el navegador, de manera que no se requiere instalación de software. La parte computacionalmente intensiva es el procesamiento de la imagen, escrito en C++ y compilado con emscripten para que se ejecute en web assembly con una performance semejante a código nativo, y las comunicaciones con websocket se manejan en javascript.

Para optimizar la performance el código en web assembly aprovecha la existencia de múltiples núcleos, ejecutándose en paralelo distribuyendo la carga en varios webworkers.

La PC ejecuta un servidor web básico escrito en Python. Este lenguaje fue elegido para facilitar las modificaciones por parte de quienes adopten esta tecnología. El servidor se comunica con el sistema de vSLAM por medio de vinculaciones (Python bindings).

El sistema de vSLAM elegido es un derivado de ORB-SLAM2 de código abierto con licencia permisiva. Escrito enteramente en C++, se ha modificado parte de su código para recibir el descriptor de imagen producido por el celular. El sistema vSLAM se compila a código nativo y se presenta como módulo Python para facilitar su interacción a quienes deseen adaptar el código para sus propios propósitos.

El trabajo profundiza en aspectos prácticos de la puesta en marcha, considerando la posibilidad de cámaras sin calibrar y proponiendo métodos pragmáticos para establecer sistemas de referencia métricos y útiles para el usuario, sobre los que se expresan las trayectorias.

En síntesis, el sistema propuesto es de código abierto, proporciona un sistema de autocalización de bajo costo que apunta a facilitar la experimentación y prototipado de AMR reales como base para el desarrollo de productos y la innovación tecnológica.

B. Principales resultados de la investigación

B.1. Publicaciones en revistas (informar cada producción por separado)

Artículo 1:	
Autores	
Título del artículo	
N° de fascículo	
N° de Volumen	



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Revista	
Año	
Institución editora de la revista	
País de procedencia de institución editora	
Arbitraje	Elija un elemento.
ISSN:	
URL de descarga del artículo	
N° DOI	



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

B.2. Libros:

Libro 1	
Autores	
Título del Libro	
Año	
Editorial	
Lugar de impresión	
Arbitraje	Elija un elemento.
ISBN:	
URL de descarga del libro	
N° DOI	

B.3. Capítulos de libros:

Autores	
Título del Capítulo	
Título del Libro	
Año	
Editores del libro/Compiladores	
Lugar de impresión	
Arbitraje	Elija un elemento.
ISBN:	
URL de descarga del capítulo	
N° DOI	

B.4. Trabajos presentados a congresos y/o seminarios:

Autores	<i>Alejandro Silvestri; Jorge Eterovic; Alesio Sinópoli</i>
Título	<i>Partición de Visual SLAM para su uso con celulares</i>
Año	<i>2023</i>
Evento	<i>XXV Workshop de Investigadores en Ciencias de la Computación</i>
Lugar de realización	<i>Junín, Pcia. Bas. As.</i>
Fecha de presentación de la ponencia	<i>Abril de 2023</i>
Entidad que organiza	<i>UNNOBA; RedUNCI</i>



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	https://wicc2023.unnoba.edu.ar/libro-de-actas/
---	---

Autores	<i>Alejandro Silvestri; Jorge Eterovic</i>
Título	<i>vSLAM Híbrido</i>
Año	<i>2023</i>
Evento	<i>11° CoNaIISI 2023 - Congreso Nacional de Ingeniería Informática / Sistemas de Información</i>
Lugar de realización	<i>San Miguel de Tucumán</i>
Fecha de presentación de la ponencia	<i>Noviembre 2023</i>
Entidad que organiza	<i>FRT – UTN; RIISIC</i>
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	https://ftrutn.cloud/conaiisi/actas-del-congreso/

Autores	<i>Alejandro Silvestri; Jorge Eterovic</i>
Título	<i>Visual SLAM híbrido para robots autónomos móviles</i>
Año	<i>2024</i>
Evento	<i>CICIC 2024. Décima Cuarta Conferencia Iberoamericana de Complejidad, Informática y Cibernética</i>
Lugar de realización	<i>Orlando, Florida, USA</i>
Fecha de presentación de la ponencia	<i>26 al 29 de Marzo de 2024</i>
Entidad que organiza	<i>International Institute of Informatics and Systemics</i>
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	<i>Resumen ACEPTADO para exponer</i>

B.5. Otras publicaciones

Autores	
Año	
Título	
Medio de Publicación	

C. Otros resultados. Indicar aquellos resultados pasibles de ser protegidos a través de instrumentos de propiedad intelectual, como patentes, derechos de autor, derechos de obtentor, etc. y desarrollos que no pueden ser protegidos por instrumentos de propiedad intelectual, como las tecnologías organizacionales y otros. Complete un cuadro por cada uno de estos dos tipos de productos.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

C.1. Títulos de propiedad intelectual. Indicar: Tipo (marcas, patentes, modelos y diseños, la transferencia tecnológica) de desarrollo o producto, Titular, Fecha de solicitud, Fecha de otorgamiento

Tipo	Titular	Fecha de Solicitud	Fecha de Emisión

C.2. Otros desarrollos no pasibles de ser protegidos por títulos de propiedad intelectual. Indicar: Producto y Descripción.

Producto	Descripción

D. Formación de recursos humanos. Trabajos finales de graduación, tesis de grado y posgrado. Completar un cuadro por cada uno de los trabajos generados en el marco del proyecto.

D.1. Tesis de grado:

Director (apellido y nombre)	Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis
Ing. Mariano Bucher	<ul style="list-style-type: none">• Alesio Esteban Abel Sinopoli• Juan Andrés Bustamante• Matías De Stefano• Agustín Baltazar Pernas• Leandro Manuel Santos Ferreiro	UNLaM	9 (nueve)	19/11/2022	UNSLAM

D.2 Trabajo Final de Especialización

Director (apellido y nombre)	Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del Trabajo Final

D.2. Tesis de posgrado: Maestría

Director (apellido y nombre)	Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

nombre)					

D.3. Tesis de posgrado: Doctorado

Director (apellido y nombre)	Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis

D.4. Trabajos de Posdoctorado

Director (apellido y nombre)	Posdoctorando (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del trabajo	Publicación

E. Otros recursos humanos en formación: estudiantes/ investigadores (grado/posgrado/ posdoctorado)

Apellido y nombre del Recurso Humano	Tipo	Institución	Período (desde/ hasta)	Actividad asignada ¹
Baltazar Pernas, Agustín	Estudiante de grado	DIIT - UNLaM	01/01/2022 a 31/12/2022	Participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM
Bustamante, Juan Andrés	Estudiante de grado	DIIT - UNLaM	01/01/2022 a 31/12/2022	Participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM
De Stefano, Matías	Estudiante de grado	DIIT - UNLaM	01/01/2022 a 31/12/2022	Participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un

¹ Descripción de la/s actividad/es a cargo (máximo 30 palabras)



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

				sistema de Visual SLAM
Santos Ferreiro, Leandro Manuel	Estudiante de grado	DIIT - UNLaM	01/01/2022 a 31/12/2022	Participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM
Sinopoli Alesio, Esteban Abel	Estudiante de grado	DIIT - UNLaM	01/01/2022 a 31/12/2022	Participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM

F. Vinculación²: Indicar conformación de redes, intercambio científico, etc. con otros grupos de investigación; con el ámbito productivo o con entidades públicas. Desarrolle en no más de dos (2) páginas.

Estamos en contacto con dos grupos de otras universidades dedicados al desarrollo de un vehículo autónomo académico. El grupo de UNLaM tuvo acceso a las experiencias en estas dos universidades, lo que permitió formular este proyecto saltando etapas anteriores.

F.1. Universidad Austral

En el departamento de electrónica de la facultad de ingeniería de la Universidad Austral, en su campus de Pilar, profesores dirigen alumnos de ing. industrial e informática en el desarrollo de vehículos autónomos en el marco de sus proyectos de grado con el que terminan su carrera.

En Austral alumnos han diseñado e implementado 5 prototipos de vehículos birrueda eléctricos con tracción a cadena y con engranajes. Para complementar esa experiencia, en UNLaM diseñamos un birrueda traccionado por correa dentada.

Adicionalmente, alumnos de Austral han implementado con éxito en 2022 un vehículo autónomo de 4 ruedas omnidireccionales, guiado por visual SLAM, usando el mismo sistema que adoptamos en UNLAM: Stella VSLAM. Para ello montaron una PC sobre el vehículo. En la UNLaM estamos yendo un paso más allá al dividir el sistema para su uso en celular.

² Entendemos por acciones de “vinculación” aquellas que tienen por objetivo dar respuesta a problemas, generando la creación de productos o servicios innovadores y confeccionados “a medida” de sus contrapartes.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Austral no tiene investigaciones en marcha en esta temática ni ha presentado estos proyectos en congresos.

F.2 Universidad de Morón

En la carrera de ingeniería electrónica de la Universidad de Morón, en el marco de Robótica y sistemas de control, profesores y alumnos han desarrollado un vehículo birrueda eléctrico, su electrónica y sus sistemas de control.

Desde 2019 los alumnos usan un birrueda en la materia de Robótica (ha sido un éxito durante la pandemia, los alumnos por turno se lo llevaron a su casa).

En este momento están llevando a cabo un proyecto de investigación para el diseño y construcción de un segundo vehículo birrueda, capaz de realizar un recorrido predefinido usando Visual SLAM como sistema de posicionamiento procesado en una PC montada en el vehículo.

G. Otra información. Incluir toda otra información que se considere pertinente.

H. Cuerpo de anexos:

- Anexo I: Copia de cada uno de los trabajos mencionados en los puntos B, C y D, y certificaciones cuando corresponda.³
- Anexo II:
 - FPI-013: Evaluación de alumnos integrantes. (si corresponde)
 - FPI-014: Comprobante de liquidación y rendición de viáticos. (si corresponde)
 - FPI-015: Rendición de gastos del proyecto de investigación acompañado de las hojas foliadas con los comprobantes de gastos.
 - FPI-035: Formulario de reasignación de fondos en Presupuesto.
- Nota justificando baja de integrantes del equipo de investigación.

³ En caso de libros, podrá presentarse una fotocopia de la primera hoja significativa o su equivalente y el índice.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Jorge Esteban Eterovic

Firma y aclaración
del director del proyecto.

Lugar y fecha : San Justo, 29 de febrero de 2024 .

- Cargar este formulario junto con los documentos correspondientes **exclusivamente** al Anexo I en SIGEVA UNLaM.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Anexo I



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Partición de Visual SLAM para su uso con celulares

Alejandro Silvestri, Jorge Eterovic, Alesio Sinopoli

Departamento de Ingeniería e Investigaciones Tecnológicas Universidad Nacional de La Matanza

Florencio Varela 1903 (B1754JEC), San Justo, (5411) 4480-8900

{[jsilvestri](mailto:jsilvestri@unlam.edu.ar); [eterovic](mailto:eterovic@unlam.edu.ar)}@unlam.edu.ar; asinopoli@alumno.unlam.edu.ar

RESUMEN

Un sistema de *Visual SLAM* obtiene en tiempo real la localización espacial de un dispositivo con cámara a partir del video que capta, en un recinto previamente mapeado. Esta capacidad se denomina **autolocalización**, imprescindible para la conducción autónoma.

Visual SLAM es una tecnología experimental que promete masificar la capacidad de autolocalización para AMR (*Autonomous Mobile Robot*), pequeños vehículos autónomos de carga para espacios controlados.

Es de sumo interés ejecutar *Visual SLAM* en celulares, pero la alta demanda computacional de estos sistemas lo hace inviable.

Este trabajo desarrolla un sistema partido de autolocalización visual basado en *Visual SLAM*, que ejecuta una parte en el celular y otra en una PC de escritorio.

Palabras clave: *VSLAM, STELLA_VSLAM, VISUAL SLAM, BIORRUEDA, AMR, ORB-SLAM*

CONTEXTO

Un AMR es un pequeño vehículo autónomo de carga, que circula muchas veces a la par de las personas, en recintos controlados - no en la vía pública - como depósitos, plantas industriales, campus y predios en general.

Los AMR comerciales usan un sistema de autolocalización basado en LIDAR, un sensor efectivo pero muy costoso.

El proyecto PROINCE - Programa de Incentivos a Docentes Investigadores

Secretaría de Políticas Universitarias (2022-2023) de la Universidad Nacional de La Matanza consiste en el desarrollo de un AMR autolocalizado por *Visual SLAM*. El proyecto es multidisciplinario, con partes mecánicas, electrónicas, de control e informáticas.

El presente trabajo se concentra en el componente informático de autolocalización del AMR que se está desarrollando en el proyecto PROINCE de la Universidad Nacional de La Matanza.

1. INTRODUCCIÓN

1.1. Sobre Visual SLAM

Sensores como *Lidar*, 2D y 3D hacen un relevamiento espacial del entorno. Montados sobre un vehículo, a medida que se desplaza sucesivas lecturas de estos sensores permiten relevar grandes áreas, llevando a cabo lo que se denomina **mapeo** del área. La composición en tiempo real de estas lecturas no es trivial, y se denomina *SLAM* (*Simultaneous Localisation And Mapping*).

Un sistema SLAM mapea en tiempo real un área mientras el vehículo se desplaza, y a la vez localiza la posición del vehículo en ese mapa. Esta capacidad de autolocalización sobre un mapa relevado es la clave para conducir de manera autónoma al vehículo a través de una trayectoria hacia un destino.

Los *Lidars* son sensores costosos. En 2007 dos ensayos fundacionales lograron los



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

primeros sistemas de *Visual SLAM*, que reemplazaron el *Lidar* por una cámara de video. Estos sistemas son PTAM [1] y *monoSLAM* [2]. En pocos años el costo de una cámara para PC se redujo enormemente haciendo posible su masificación, poniendo a *Visual SLAM* como una tecnología candidata a sustituir los sistemas de SLAM con *Lidar*.

A estos trabajos pioneros de *Visual SLAM* le siguieron una sucesión de papers que marcaron una evolución acelerada durante 10 años. En 2014 y 2015 los ensayos LSD-SLAM [3] y ORB-SLAM [4] establecieron el hito máximo en cuanto a performance y precisión, sin alcanzar el “santo grial”: la posibilidad de poder ejecutarse en un celular.

En 2017 *Visual SLAM* se consideró un trabajo terminado, con ORB-SLAM2 [5] representando el estado del arte hasta la fecha. Varios ensayos posteriores han contribuido en mejorar diversos aspectos, entre los que se destacan el uso de cámara estéreo, omnidireccional, RGBD y combinación con acelerómetros, pero con apenas mejoras menores en la performance.

Cabe mencionar el surgimiento de una rama de investigación en 2018, denominada *Spatial AI* [6], que procura desarrollar sistemas de *Visual SLAM* con inteligencia artificial. Esta rama, con muchos méritos propios, no ha aportado a la mejora de la performance (sino todo lo contrario), divergiendo significativamente del objetivo de este trabajo.

En 2019 el ensayo *OpenVSLAM* [7] puso a disposición de la comunidad un sistema de *Visual SLAM* basado en ORB-SLAM2 pero con código abierto con licencia BSD (uso libre e irrestricto), con un código totalmente reescrito aplicando buenas prácticas de programación, y corrigiendo una infinidad de bugs que impiden el uso práctico de los otros sistemas publicados.

En 2021 *OpenVSLAM* se retiró por presuntas violaciones de licencias, y fue sustituido por *stella_vslam* [8], un proyecto abierto dedicado a corregir todas las posibles violaciones de licencia de *OpenVSLAM*, convirtiéndose en la

única solución de código abierto para proyectos de ingeniería y productos comerciales.

1.2 Visual SLAM con celulares

Hay dos antecedentes de uso de *Visual SLAM* en celulares, ninguno de los cuales satisface la necesidad de un AMR.

En 2015 LSD-SLAM demostró que puede ser usado en celulares para autocalización sobre mapas relevados, desactivando el módulo de mapeo que es el de mayor demanda computacional. A priori esto satisface la necesidad de autocalización para AMR, pero falla en otros aspectos, principalmente en el enorme tamaño de los mapas en memoria, y en la incapacidad de guardarlos en un archivo para ser reutilizados en otro momento.

Entre 2018 y 2023 se han repetido con éxito varios intentos de compilación de ORB-SLAM2 (entre otros) en celulares de alta gama, pero obteniendo siempre una performance muy pobre.

La idea de partir el procesamiento entre un celular con cámara montado en el AMR y una PC fija conectada por WIFI claramente no elimina la necesidad de la PC, pero flexibiliza enormemente el despliegue del sensor (la cámara) en el vehículo experimental, habilitando incluso la experimentación con plataformas muy baratas como los pequeños vehículos robot Arduino.

El planteo simplista consiste en usar el celular como cámara IP y transmitir en tiempo real el video por WIFI a la PC con *Visual SLAM*. Lamentablemente la compresión del video imprescindible para acomodarse en el ancho de banda disponible introduce un retardo superior a 1 s que hace inviable su uso en navegación autónoma.

En Diciembre de 2022 alumnos de grado de ingeniería informática de la Universidad Nacional de La Matanza, participando en el proyecto PROINCE, presentaron con éxito su trabajo de grado UNSLAM [9], la primera aplicación con *Visual SLAM* partido, con



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

performance adecuada para la navegación autónoma de un AMR.

Este trabajo demuestra experimentalmente la viabilidad del concepto, logrando un retardo de 200 ms en un celular antiguo de bajo poder computacional, suficiente para la navegación autónoma, y muy mejorable con el uso de celulares de más potentes.

Más notablemente, el sistema no se implementó en una App, sino en una página web, lo que le confiere amplia compatibilidad y fácil despliegue.

UNLaM es la prueba de concepto del módulo de navegación del proyecto PROINCE. Adopta *stella_vslam* como sistema de *Visual SLAM* ejecutando en una PC.

El *pipeline* de *stella_vslam* es complejo, pero a los efectos de la exposición este trabajo lo dividió en dos partes: el preprocesamiento en celular de la imagen obtenida por la cámara, y el postprocesamiento en PC de la información extraída al combinarse con el mapa.

El preprocesamiento es el conjunto de todas las operaciones que se realizan sobre la imagen. Al finalizar, la imagen se descarta, y el sistema sólo utiliza la información extraída de ella, que ocupa una fracción menor que la imagen completa.

Esta es la parte que se ejecuta en el celular:

1. captación de la imagen color y conversión a monocromática
2. generación de pirámide
3. detección de *Keypoints* [10] en todos los niveles de la pirámide
4. selección de *Keypoints* representativos por medio de *quadtree*.
5. extracción de descriptores [11, 12]

Este preprocesamiento está desarrollado en C++ utilizando la biblioteca *OpenCV* [13]. Su implementación en una página web requirió:

- aislar los módulos de *OpenCV* utilizados
- compilarlos como bibliotecas estáticas

- adoptar un soporte de paralelismo *OpenMP* [14] con una biblioteca para *Web Assembly*
- compilar el código C++ del pipeline para *Web Assembly* con *emscripten* [15] y *llvm* [16], vinculando las bibliotecas estáticas
- elaborar la página web que capte la imagen con *javascript* y la preprocese con el módulo anterior en *Web Assembly*
- enviar los resultados al servidor: la opción directa, *fetch* y *JSON*, se descartó para favorecer una más performante que evita el *parseo*: *web sockets* para enviar los datos de manera binaria

De lado de la PC se desarrolló una variante de *stella_vslam* que admite como entrada los datos preprocesados por el celular en lugar de una imagen.

Además se implementaron *bindings* de *stella_vslam* para Python, con dos propósitos principales:

- permitir al servidor web que recibe los datos del preprocesamiento, enviarlos al *stella_vslam*
- obtener el resultado buscado: la pose de localización

1.3. Sistema de referencia real

En general los sistemas de *Visual SLAM* presentan la información de pose en términos de una matriz de 4x4 en coordenadas homogéneas que representan una transformación euclidiana 3D.

Durante el mapeo los sistemas de *Visual SLAM* establecen un sistema de referencia espacial virtual, con un centro y una orientación de ejes arbitraria y desconocida por el usuario. En particular, los sistemas monoculares - aquellos que usan una sola cámara, en contraste con los estereoscópicos y los de profundidad - adoptan también una escala arbitraria, ya que carecen de referencia de escala real.

La segunda parte del presente trabajo fue diseñar e implementar una estrategia de vinculación entre el sistema de referencia virtual arbitrario del *Visual SLAM*, y un



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

sistema de referencia real definido por el usuario. Los pasos para la vinculación de estos sistemas de referencia son:

1. el usuario posiciona el celular en el punto que se elige como origen (0;0;0) del sistema de referencia, que suele ser en el suelo; el sistema registra las coordenadas virtuales asociadas obtenidas por Visual SLAM, y el acelerómetro del celular para relevar la dirección vertical
2. a continuación el usuario posiciona el celular desplazándolo 1 m sobre el eje X, a las coordenadas (1;0;0); el sistema registra esas coordenadas virtuales y la dirección vertical

Se asume que el suelo es horizontal, el eje Y es vertical y el eje Z es perpendicular a los otros dos.

Con las dos mediciones el algoritmo calcula la matriz de transformación que convierte del sistema virtual arbitrario de *Visual SLAM* al sistema métrico real elegido por el usuario.

Durante la operación, para cada imagen procesada el sistema de *Visual SLAM* informa sus coordenadas virtuales, y el código propio usa la matriz anterior para convertirlas a coordenadas métricas.

2. LÍNEAS DE INVESTIGACION Y DESARROLLO

Este trabajo es parte del proyecto PRINCE de la Universidad Nacional de La Matanza, dedicado al diseño de un AMR versátil de bajo costo como prototipo para uso académico y para uso comercial.

3. RESULTADOS OBTENIDOS/ESPERADOS

Los resultados obtenidos se describieron en la Introducción, y se resumen aquí:

- prueba de concepto de Visual SLAM partido, con una medición de performance adecuada para AMR

- método de vinculación entre sistemas de referencias virtual y real
- el sistema UNLaM completo y funcionando publicado en GitHub

Los resultados esperados son:

- Aislación del autocalizador partido como un módulo para su disponibilidad en otros proyectos, incluyendo PROINCE
- Modificación y documentación del módulo
- Tabla comparativa con pruebas de performance con varios celulares
- Sistema de navegación que indique la trayectoria a seguir a partir de la autocalización.

4. FORMACIÓN DE RECURSOS HUMANOS

Cinco alumnos de ingeniería informática se recibieron de ingenieros con su trabajo de grado UNLaM.

El proyecto promueve la formación en proyectos de I+D de 4 docentes ingenieros.

5. BIBLIOGRAFIA

[1] PTAM, Parallel Tracking and Mapping for Small AR Workspaces, Klein-Murray, ISMAR. 2007.

[2] MonoSLAM: Real-Time Single Camera SLAM, Davidson et al, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 29, NO. 6, JUNE 2007.

[3] LSD-SLAM: Large-Scale Direct Monocular SLAM, Engels et al, Technical University Munich. 2014.

[4] ORB-SLAM: a Versatile and Accurate Monocular SLAM System, Mur Artal et al, IEEE TRANSACTIONS ON ROBOTICS. 2015.

[5] ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Cameras, Mur Artal et al, IEEE TRANSACTIONS ON ROBOTICS. 2017.

[6] ~~FutureMapping~~: The Computational Structure of Spatial AI Systems, Davidson et al. 2018.

[7] ~~OpenVSLAM~~: A Versatile Visual SLAM Framework, Sumikura et al, Arxiv. 2019.

[8] https://github.com/stella-cv/stella_vslam

[9] <https://github.com/UNLaM>

[10] Faster and better: a machine learning approach to corner detection, Rosten et al, 2008.

[11] BRIEF: Binary Robust Independent Elementary Features, Calonder et al, 2011.

[12] ORB: an efficient alternative to SIFT or SURF, Rublee et al, 2012.

[13] <https://docs.opencv.org/4.7.0/>

[14] <https://www.openmp.org/>

[15] <https://emscripten.org/>

[16] <https://llvm.org/>



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



**UNIVERSIDAD NACIONAL
NOROESTE BUENOS AIRES**
ESCUELA DE TECNOLOGÍA



Se certifica que Eterovic Jorge E., ha participado en calidad de Asistente en el XXV Workshop de Investigadores en Ciencias de la Computación (WICC 2023), realizado los días 13 y 14 de Abril de 2023 en la Ciudad de Junín, Buenos Aires, Argentina.




Lic. Patricia Pesado
Coordinadora Titular
Red Unci




Lic. Lucas Benjamin
Secretario Académico Escuela de Tecnología
Jun 29/16/2023, 18:14
Junín, Argentina
Director ITT




Lic. Monica Sarobe
Directora de la Escuela de Tecnología
UNNOBA



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

vSLAM Híbrido

Alejandro Silvestri, Jorge Eterovic
Departamento de Ingeniería e Investigaciones Tecnológicas
Universidad Nacional de La Matanza
{jasilvestri; eterovic}@unlam.edu.ar

Resumen

Una característica esencial que brinda autonomía a los robots móviles autónomos es su capacidad de autolocalización precisa. La tecnología emergente SLAM visual reduce sustancialmente el costo del sensor, reemplazando un costoso LIDAR por una cámara común. Con la intención de facilitar la experimentación con SLAM visual en robots móviles de bajo costo (incluyendo kits Arduino), este proyecto propone la ejecución de un sistema de SLAM visual en el navegador, para ser usado en celulares sin necesidad de instalación; y delegando parte de la carga computacional al backend, de modo que pueda ser usado con celulares de media y baja gama.

Introducción

Los robots móviles autónomos (AMR) han surgido en la década pasada como producto para logística, predominantemente en depósitos de plantas de manufactura. Están sustituyendo gradualmente a los AGV (*automated guided vehicles*) que requieren desplegar in situ guías para sus trayectorias predeterminadas. Los AMR usan un sensor *LIDAR* para localizarse en un mapa de las instalaciones, y luego seguir trayectorias trazadas virtuales, evitando el uso de guías.

La tecnología de SLAM visual, o vSLAM, ha alcanzado la madurez en laboratorio en 2017 con la publicación del paper con código abierto ORB-SLAM2[5], y se encuentra en el lento camino de desarrollo de producto. vSLAM sustituye el costoso sensor *LIDAR* por una cámara común, prometiendo reducir enormemente el costo de la tecnología de autolocalización, habilitando nuevos tipos y usos de AMR.

[Al día de hoy](#) para implementar un robot móvil autónomo en laboratorio se requiere montar una PC en el vehículo, capaz de realizar el pesado cómputo de estos algoritmos. Nuestro novedoso sistema híbrido distribuye el trabajo entre una PC backend y una página web en el navegador de un celular que puede ser montado en cualquier vehículo eléctrico de laboratorio, incluyendo kits Arduino, lo que hace más accesible la experimentación con AMR.

Evolución de vSLAM y estado del arte

SLAM (simultaneous localization and mapping) es una técnica para sensores de distancia precisa como *LIDARs*, que determina la localización de un móvil mediante el alineamiento de las observaciones del sensor con las observaciones previas registradas en un mapa que representa el conocimiento del entorno, a la vez que extiende progresivamente el mapa incorporando las observaciones de zonas inexploradas. Estas dos acciones, denominadas localización y mapeo, se realizan simultáneamente para cada muestreo del entorno.

En 2007 dos papers fundacionales, monoSLAM[2] y PTAM[1], presentaron los primeros sistemas de SLAM visual o vSLAM, llevando a cabo un *SLAM* en tiempo real pero con el uso de una cámara en lugar de *LIDAR*. Conceptualmente, dos imágenes de la misma escena obtenidas desde posiciones diferentes permiten triangular objetos obteniendo su posición 3D, proveyendo las observaciones de distancia requeridas por *SLAM*.

En los años siguientes se disparó el interés por el desarrollo de sistemas de vSLAM, traccionados por su prometedora aplicación en el naciente campo de la realidad aumentada. Esta tecnología experimentó una rápida evolución a través de un proceso de prueba y error, en el que se intentaron y descartaron innumerables abordajes, prevaleciendo estos tres principales:

- indirecto, disperso o “de partículas”, el más difundido
- directo o denso, más lento
- semidenso

Para 2015 los sistemas LSD-SLAM[3] (semidenso) y ORB-SLAM[4] (disperso) alcanzaron alta precisión con muy bajos recursos computacionales, mucho menores a sus predecesores incluyendo los sistemas fundacionales, pudiendo ejecutarse en PC común sin necesidad de GPU.

En 2017 ORB-SLAM2 ganó el premio al mejor paper del año y se convirtió en hito de la maduración tecnológica de vSLAM, proveyendo acceso a su código abierto, ganando competencias y superando a sus rivales tanto en precisión como en performance. Si bien la performance superior no alcanzó para su ejecución en celulares, vSLAM se consideró un problema resuelto, declinando el interés por



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

mejorar su performance. La comunidad se abocó a resolver problemas periféricos:

- variedad de tipos de cámaras (gran angular, omnidireccional, estéreo, RGBD)
- *visual inertial SLAM* y *fusion vSLAM*, incorporando IMU y otros sensores
- odometría visual
- unificación de múltiples mapas
- *Spatial AI[5]*: migración de vSLAM a plataformas de *deep learning*

Muchos sistemas de vSLAM posteriores se basaron en ORB-SLAM2. En particular en 2018 OpenVSLAM[7] replanteó su código por completo con mejores prácticas de desarrollo, logrando un sistema más robusto que funciona por tiempos prolongados sin colgarse, ganando el premio al mejor sistema de código abierto de ese año. Los autores de OpenVSLAM fueron demandados por violaciones a la licencia GPL de ORB-SLAM2, el proyecto fue rescatado por *stella-vslam[8]*, un repositorio de la comunidad que continúa su desarrollo de código abierto sin publicar papers.

Dispositivos móviles

Los sistemas de vSLAM tienen una demanda computacional que en 2017 superaban las capacidades de los teléfonos celulares para sus aplicaciones generales. Si bien los celulares han ido aumentando su capacidad de cómputo, ésta se ha desarrollado principalmente en GPU y TPU para soporte de gráficos y modelos de *deep learning*. Los sistemas de vSLAM no emplean estos recursos, y no se vieron tan beneficiados por esta evolución. A la fecha un sistema de vSLAM completo de código abierto puede funcionar mínimamente en un dispositivo de alta gama, pero no funciona en la gran mayoría de los celulares.

En este contexto definimos dispositivo móvil abarcando también aquellos que no son teléfonos, como por ejemplo las placas de desarrollo Raspberry Pi y Nvidia Jetson TX1. A continuación se ilustra el estado del arte de esta tecnología en dispositivos móviles:

- ARCore de Google[13], ARKit de Apple, totalmente orientados a la realidad aumentada en celulares
- SLAMCore[16] lanzó en 2020 un kit de vSLAM inercial para cámara estereoscópica, que ejecuta en tiempo real en Raspberry Pi 4 (y en otras más poderosas)

SLAMCore es una empresa británica con fines de lucro fundada por los iniciadores de vSLAM; su desarrollo es propietario y secreto; y su kit no está disponible para instituciones educativas, de manera que no representa una opción real, aunque sí demuestra el estado del arte.

Pipeline de vSLAM

Los sistemas de vSLAM indirectos tienen una secuencia de procesamiento similar, comenzando por la **extracción de características** de la imagen. Las imágenes capturadas son

utilizadas exclusivamente en esta etapa. El resto del sistema opera exclusivamente sobre las características extraídas, prescindiendo totalmente de las imágenes.

A continuación se describe la etapa de extracción de características de ORB-SLAM, que es adoptada por todos sus derivados incluyendo ORB-SLAM2, ORB-SLAM3, OpenVSLAM y *stella-vslam*:

1. captura de la imagen
2. conversión a monocromática
3. expansión de la imagen en pirámides
4. detección de keypoints
5. homogeneización espacial con quadtree
6. cómputo de la orientación de las características
7. extracción de descriptores orientados



vSLAM utiliza estas características de diferente manera dependiendo del estado del sistema:

- inicialización, mapeo entre dos imágenes para una evaluación homográfica y otra epipolar; obtenida la pose relativa se triangulan los puntos 3D que conforman el mapa inicial
- localización o *tracking*, mapeo entre los descriptores de la imagen actual y los del mapa, evaluando Perspective from N Points (específicamente EPnP) para obtener la pose de la cámara en función de las observaciones
- mapeo, mapeo entre la imagen actual e imágenes previas registradas en *keyframes*, para la triangulación de nuevos puntos
- relocalización, mapeo entre los descriptores de la imagen actual y los del mapa

Ésta es una enumeración superficial de un proceso mucho más complejo en el que no se ahonda porque este trabajo se concentra en la etapa de extracción de características.

Propuesta

Para utilizar un celular como sensor inalámbrico de un sistema de vSLAM que se ejecuta en una PC, la primera idea consiste en utilizarlo como cámara WIFI. Los dos



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

inconvenientes inmediatos proceden de la compresión de video para la transmisión:

- la compresión pierde calidad de imagen, lo que conlleva a una pérdida de precisión en la localización
- la compresión introduce un retardo en la transmisión del orden de los 2s¹, que lo hace inservible para la navegación autónoma

Siguiendo el mismo razonamiento, la segunda propuesta consiste en transmitir sin compresión la imagen capturada de la cámara. Esto es viable reduciendo el flujo a un mínimo aceptable de 5 imágenes por segundo, con una resolución de 640 x 480 píxeles. Este abordaje funciona pero consume mucho ancho de banda, usualmente no disponible con una conexión WIFI de uso compartido.

Este trabajo hace uso del concepto de *edge computing*, llevando el trabajo de **descripción** al celular, específicamente a una página web, y el resto en una PC. A este sistema lo denominamos vSLAM híbrido.

vSLAM híbrido

El sistema separa el procesamiento en dos partes:

- **frontend**, que se ejecuta en el navegador de un celular y se ocupa de la **descripción** (los primeros 7 pasos enumerados) desde la captura de la imagen con la cámara y produce un descriptor de la imagen
- **backend**, un servidor web que sirve al frontend, recibe de él los descriptors vía websockets, y los envía al sistema vSLAM modificado para este propósito

El sistema vSLAM elegido fue *stella-vslam*, de código abierto escrito íntegramente en C++. La primera tarea consistió en su ingeniería inversa, identificando la secuencia de descripción. Una vez identificada se modularizó y se separó del resto del sistema, aislando así este módulo de descripción para ser portado al navegador.

Frontend

El frontend es una página web que se puede ejecutar en cualquier navegador. A los efectos prácticos en este documento se asume que se ejecutará en el navegador de un celular. Para el ciclo de testeo resultó muy práctico poder ejecutarlo en el navegador de la misma PC donde se desarrolla.

La demanda computacional hace inútil la transcripción a Javascript del módulo de descripción. El módulo debe ejecutarse en WebAssembly para lograr una performance cercana a código nativo. Para lograrlo se llevaron a cabo las siguientes tareas:

1. identificación de las bibliotecas de OpenCV requeridas por el módulo, para luego compilarlas en WebAssembly como bibliotecas estáticas `.a`; y activando la capacidad de computación paralela para mayor performance

¹ Varía según la capacidad del celular

2. elaboración de **bindings** de Javascript, para que las funciones principales puedan invocarse desde el código javascript, enviando la imagen y rescatando el descriptor resultante
3. compilación con `emscripten`, usando el compilador `clang` que, a diferencia de `g++`, puede compilar a WebAssembly

El resultado es un módulo `wasm` que recibe una imagen en forma de `Uint8Array` y devuelve su descriptor bajo la misma forma.

El resto del frontend es trivial: además de la interfaz de usuario, el código javascript debe capturar una imagen del feed de cámara para enviarlo al módulo de descripción en forma de `Uint8Array`. El descriptor de imagen obtenido se envía al backend vía `websockets` en forma de `ArrayBuffers`.

Backend

El backend se desarrolló en Python y consta de tres partes:

- simple http server, encargado de servir la página web
- servidor de `websocket`, que recibe el descriptor de imagen desde el celular
- módulo de vSLAM vinculado a Python, al que se entrega el descriptor de imagen y devuelve la localización

Las dos primeras partes requieren desarrollo común. El módulo de vSLAM demandó desarrollo profundo para lograr estos dos objetivos:

- modificar el pipeline de entrada, por donde usualmente se alimenta una imagen, para que acepte características extraídas de la imagen y se saltee la etapa de extracción que ya fueron ejecutados en el celular, y compilar con `g++` para producir un `shared object .so`
- vincular métodos de vSLAM con Python, lo que se hace escribiendo código C++ y utilizando la biblioteca `PyBind11`, de manera que al compilar con `g++` produce otro `shared object` que además cumple con las reglas de `Cython` y se puede importar como módulo de Python

De esta manera, el módulo de vSLAM se importa en Python, desde donde se crean los objetos y se los configura, para luego alimentar de a uno los descriptors de imágenes que se reciben desde el celular. El método da como resultado una matriz de pose que representa la localización de la cámara en el mundo virtual del sistema vSLAM.

Matriz de pose

Los sistemas de vSLAM expresan la pose de la cámara con una matriz de 4x4 que representa una transformación euclidiana[9] de un espacio 3D, expresada en su espacio proyectivo asociado. La matriz está normalizada, de manera que la última fila siempre es (0; 0; 0; 1). Esta



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

transformación, denominada T_{cw} , convierte coordenadas del espacio del mundo virtual (w) a coordenadas del sistema de referencia de la cámara (c). La pose de la cámara T_{wc} se obtiene con:

$$T_{wc} = T_{cw}^{-1}$$

La posición ($x, y, z, 1$)^T de la cámara es la última columna de T_{wc} .

Tecnologías

A continuación se listan las tecnologías involucradas que fue necesario comprender y utilizar para llevar a cabo este trabajo. En lo tocante a código y vinculaciones:

- C++ y g++, para modificar *stella-vslam* y establecer *bindings*
- *emscripten*^[12] para compilar código C++ a WebAssembly, y para establecer los *bindings* entre Javascript y C++
- *PyBind11*^[13], biblioteca para C++ de *bindings* de Python, para generar un módulo de código nativo importable en Python
- Javascript, CSS, HTML para el frontend
- *Uint8Array* como contenedor llano para el intercambio de información entre Javascript y WebAssembly
- *Websockets* para las comunicaciones
- Python para backend

En lo referente a visión artificial:

- Biblioteca *OpenCV*^[14] (open computer vision)
- Manipulación de imágenes, conversión a monocromático
- Pirámides: generación de imágenes de menor de resolución para invarianza a la escala
- Clase *Keypoint*, que se debe serializar como parte del resultado
- detector de keypoints FAST
- *quadTree*, segmentación espacial para limitar la cantidad de keypoints manteniendo una distribución espacial uniforme
- ORB[12]: cálculo de orientación y extracción de descriptores BRIEF[11]

En lo referente a pose de vSLAM:

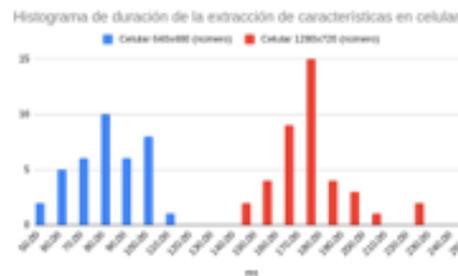
- Coordenadas homogéneas
- Transformación euclídeana
- Cambio de sistema de referencia por medio de una transformación lineal

Resultados

A continuación se muestran algunas mediciones sobre el sistema funcionando.

Performance

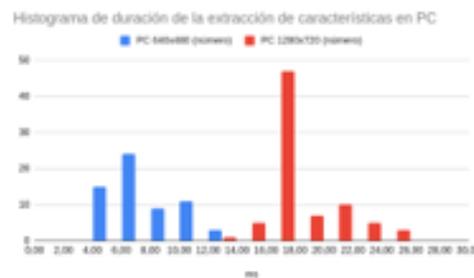
Se han hecho mediciones de la duración de la extracción de características en navegador accediendo a la cámara con dos resoluciones diferentes: 640x480px y 1280x720px. Interesan principalmente las mediciones del sistema ejecutando en celular. Se ha empleado un celular de gama media Motorola One Fusion:



Para 640x480px se observa una moda de 80 ms, lo que permite procesar 12 fps (imágenes por segundo). Para 1280x720px la moda es 180 ms, apto para 5 fps.

Pruebas con celulares antiguos y de baja gama demostraron superar ampliamente los 5 fps para 640x480px.

El siguiente gráfico muestra el rendimiento en una PC, a modo meramente ilustrativo:



En este caso las modas se encuentran en 6 ms (160 fps) para 640x480px, y 18 ms (55 fps) para 1280x720px.

El código WebAssembly ejecuta tantos *webworkers* como núcleos tenga el procesador. Es la variante para navegador de la técnica conocida como *pool de threads* que se basa en que la máxima paralelización aprovechable consiste en ejecutar un *thread* por núcleo. Con el inspector del navegador se verifica la existencia de estos *webworkers* y su carga durante el procesamiento de la imagen: el código efectivamente se paraleliza cuando tiene la oportunidad de hacerlo.

Ancho de banda

En cada imagen se detectan en el orden de 1.000 puntos singulares (este valor depende de la configuración). La



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

extracción de características de la imagen consiste en la extracción de las características de estos 1.000 puntos singulares.

A cada punto singular detectado en la imagen le corresponde un objeto KeyPoint y un Descriptor, que constituyen el conjunto de características requeridas. El descriptor de ORB es binario de 256 bits (32 bytes), sin posibilidad de compresión. El objeto KeyPoint consta de 7 propiedades de 32 bits (int o float). Un análisis profundo del código del sistema vSLAM reveló que a pesar de guardar las 7 propiedades, sólo utiliza 4 de ellas. La serialización del KeyPoints se reduce a estas 4 propiedades de 32 bits cada una:

- pt.x
- pt.y
- octave
- angle

Estas 4 propiedades ocupan 16 bytes, que sumados a los 32 bytes del descriptor totalizan 48 bytes por punto singular. Para una extracción de 1.000 puntos singulares, el conjunto de características ocupa 48 KB, lo que supone una enorme reducción comparado con el tamaño de la imagen. Una imagen de 1280x720px ocupa 920KB.

Comparación de ancho de banda entre características procesadas e imágenes crudas

	1.000 características	Imagen monocromática de 1280x720px
Tamaño	48 KB	920KB
Ancho de banda para 10 fps	4 Mbps	74 Mbps

Conclusiones

Este sistema abre la posibilidad del uso de vSLAM con celulares de gama baja y media, facilitando la experimentación con tecnología de vehículos autónomos, acercándola a universidades e incluso al nivel secundario.

El paso siguiente sería empaquetar el sistema para facilitar su distribución e instalación.

Referencias

[1] PTAM, Parallel Tracking and Mapping for Small AR Workspaces, Klein-Murthy, ISMAR 2007

[2] MonoSLAM| Real-Time Single Camera SLAM, Davidson et al, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 29, NO. 6, JUNE 2007

[3] LSD-SLAM: Large-Scale Direct Monocular SLAM, Engels et al, Technical University Munich 2014

[4] ORB-SLAM: a Versatile and Accurate Monocular SLAM System, Mur Aral et al, IEEE TRANSACTIONS ON ROBOTICS 2015

[5] ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras, Mur Aral et al, IEEE TRANSACTIONS ON ROBOTICS 2017

[6] FutureMapping: The Computational Structure of Spatial AI Systems, Davidson et al 2018

[7] OpenVSLAM: A Versatile Visual SLAM Framework, Simikuz et al, Arxiv 2019

[8] https://github.com/stella-cv/stella_vslam

[9] Hartley, Richard and Zisserman, Andrew, "Multiple View Geometry in Computer Vision", Cambridge University Press, 2004, pp. 38-39.

[10] Faster and better: a machine learning approach to corner detection, Rosten et al, 2008

[11] BRIEF: Binary Robust Independent Elementary Features, Calonder et al, 2011

[12] ORB: an efficient alternative to SIFT or SURF, Rublee et al, 2012

[13] Google ARCore [online] <https://developers.google.com/ar>

[14] OpenCV. (2023). [Open-Source Computer Vision Library](https://docs.opencv.org/4.7.0/). [Online]. <https://docs.opencv.org/4.7.0/>

[15] Emscripten [online] <https://emscripten.org/>

[16] SLAMCore [online] <https://www.slamcore.com/>

[17] PyBind11 [online] <https://pybind11.readthedocs.io/en/stable/>

[18] Partición de Visual SLAM para su uso con celulares, Silvestri, A. y Eterovic, J., Universidad de La Matanza, WICC 2023

[19] UNSLAM [online] <https://github.com/UNSLAM>



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Visual SLAM híbrido para robots autónomos móviles Resumen extendido para CICIC 2024

Alejandro SILVESTRI

Departamento de ingeniería e investigaciones tecnológicas, Universidad Nacional de La Matanza
La Matanza, Buenos Aires, B1754, Argentina

Jorge Eterovic

Departamento de ingeniería e investigaciones tecnológicas, Universidad Nacional de La Matanza
La Matanza, Buenos Aires, B1754, Argentina

RESUMEN EXTENDIDO:

Un robot móvil autónomo (*AMR, Autonomous Mobile Robot*) es un vehículo autónomo para predios, interiores y otros lugares controlados. A diferencia de los automóviles autónomos (*self-driving cars*), los AMR no cuentan con un camino claramente visible; tampoco cuentan con marcas de ruta ni con la asistencia de balizas. En su lugar tienen en memoria un mapa y la trayectoria que deben seguir, y emplean un sensor para una localización precisa. Los AMR comerciales adoptan costosos sensores *LiDAR* cuya salida es procesada por sistemas *SLAM (Simultaneous Localization And Mapping)* computacionalmente intensivos que les permiten deducir su localización a partir de un relevamiento de los alrededores del robot. Tanto el *LiDAR* como la computadora que procesa *SLAM* se encuentran a bordo del robot.

Una alternativa a la dupla *LiDAR* y *SLAM* son los sistemas de *SLAM* visual o *vSLAM*, que emplean una cámara de video como sensor. La tecnología *vSLAM* sigue siendo experimental pero está alcanzando un alto grado de madurez. Estos sistemas evitan el elevado costo de un *LiDAR*, reemplazándolo por una cámara de bajo costo. Los sistemas *vSLAM* son computacionalmente intensivos y no pueden ejecutarse en un celular, por lo que un AMR con *vSLAM* requiere una computadora a bordo.

Con la intención de evitar montar una computadora a bordo, se ha estudiado la posibilidad de montar una cámara *WiFi* (que se puede implementar con un celular), pero presenta estos serios inconvenientes:

1. El flujo de video comprimido tiene un retardo significativo, superior a un segundo, y como la compresión tiene pérdidas la calidad de la imagen se degrada, reduciendo la capacidad de localización.
2. Un flujo de imágenes sin comprimir ocupa demasiado ancho de banda, usualmente no disponible en las redes *WiFi*.

Este artículo propone un sistema de *vSLAM* partido, con un celular a bordo que capture las imágenes y realice un preprocesamiento, extrayendo el denominado *descriptor de imagen*, para enviarlo por *WiFi* a una computadora estática, fuera del robot, que ejecuta la parte restante del sistema *vSLAM* y obtiene la localización del robot. El descriptor de imagen tiene un tamaño en bytes significativamente menor que el de la propia imagen, y las pruebas confirman que se pueden enviar por *WiFi* sin inconvenientes.

Este sistema híbrido adopta un celular como dispositivo de borde (*edge computing*), fácil de montar de una variedad de robots, habilitando la capacidad de localización visual para robots pequeños y de prototipo, como las aspiradoras autónomas y los robots *Arduino*, y permitiendo ejecutar *vSLAM* en una computadora de escritorio o incluso en la nube.

La división de tareas elegida habilita el uso de celulares de baja gama, lo que facilita la adopción de esta tecnología. Las pruebas con celulares de muy baja gama logran procesar 5 imágenes por segundo, suficiente para la conducción autónoma de los AMR.

Para mayor compatibilidad el procesamiento en celular se realiza enteramente en una página web en el navegador, de manera que no se requiere instalación de software. La parte computacionalmente intensiva es el procesamiento de la imagen, escrito en *C++* y compilado con *emscripten* para que se ejecute en *web assembly* con una performance semejante a código nativo, y las comunicaciones con *websocket* se manejan en *javascript*.

Para optimizar la performance el código en *web assembly* aprovecha la existencia de múltiples núcleos, ejecutándose en paralelo distribuyendo la carga en varios *webworkers*.

La *PC* ejecuta un servidor web básico escrito en *Python*. Este lenguaje fue elegido para facilitar las modificaciones por parte de quienes adopten esta tecnología. El servidor se comunica con el sistema de *vSLAM* por medio de vinculaciones (*Python bindings*).

El sistema de *vSLAM* elegido es un derivado de *ORB-SLAM2* de código abierto con licencia permisiva. Escrito enteramente en *C++*, se ha modificado parte de su código para recibir el descriptor de imagen producido por el celular. El sistema *vSLAM* se compila a código nativo y se presenta como módulo *Python* para facilitar su interacción a quienes deseen adaptar el código para sus propios propósitos.

El artículo profundiza en aspectos prácticos de la puesta en marcha, considerando la posibilidad de cámaras sin calibrar y proponiendo métodos pragmáticos para establecer sistemas de referencia métricos y útiles para el usuario, sobre los que se expresan las trayectorias.

En síntesis, el sistema propuesto es de código abierto, proporciona un sistema de autocalización de bajo costo que apunta a facilitar la experimentación y prototipo de AMR.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

reales como base para el desarrollo de productos y la innovación tecnológica.

Palabras Clave: SLAM visual, ORB-SLAM2, computación de borde, computación en la nube, Robots Móviles Autónomos, Vehículos autónomos, C++, Python, PyBind11, websockets, emscripten, web assembly.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	6
Vigencia	03/9/2019

CICIC 2024 - Aceptación de Artículo

cicic-sec@mail.iiis-spring24.org <cicic-sec@mail.iiis-spring24.org>

Mié 31/01/2024 13:16

Para: JORGE ETEROVIC <eterovic@unlam.edu.ar>

Estimado(a) Ing. Jorge ETEROVIC,

Nos dirigimos a usted en relación al artículo que propuso a CICIC 2024, titulado: "Visual SLAM híbrido para robots autónomos móviles" (CB701YQ), cuya notificación de aceptación le fue enviada el 13 de diciembre de 2023 (ver e-mail anexo abajo), ya que a la fecha no tenemos registro de su inscripción en la conferencia al cual esté asociado dicho artículo.

Si usted ya se registró en la conferencia con este artículo, pero con un código de identificación diferente, o si usted olvidó ingresar el código de identificación del artículo cuando se registró, por favor, informenos al respecto.

Si usted no recibió el e-mail en donde se le notificaba la aceptación de su artículo, anexo abajo, y la fecha tope para el envío de la versión final del mismo y para optar al descuento por inscripción anticipada ha pasado, no tenemos inconveniente alguno en extendersele al **27 de febrero de 2024**.

Si usted desea retirar su artículo, mucho sabríamos agradecerle el que no los haga saber.

Si usted cree que recibió este e-mail por error, le expresamos nuestras disculpas y le agradecemos que nos los comuniqué a objeto de tomar las acciones correctivas respectivas.

Atentamente,

Secretariado de CICIC 2024

----- Original Message -----

From: <cicic-system@mail.iiis-spring24.org>

To: <Eterovic@unlam.edu.ar>

Sent: 13 de diciembre de 2023 9:07:06 AM

Subject: CICIC 2024 - Aceptación de Artículo

***** POR FAVOR, NO ENVÍE CORREOS ELECTRÓNICOS A ESTA DIRECCIÓN *****

Esta dirección es una cuenta del sistema automatizado y su buzón no es revisado, por lo que los correos electrónicos enviados a la misma no podrán ser respondidos.

Estimado(a) Ing. Jorge ETEROVIC:

En nombre del Comité Organizador de la Décima Cuarta Conferencia Iberoamericana de Complejidad, Informática y Cibernética (CICIC 2024), que tendrá lugar del 26 al 29 de marzo de 2024, tengo el agrado de informarle que, en base a las recomendaciones hechas por los revisores del resumen extendido que usted envió, su artículo titulado: "Visual SLAM híbrido para robots autónomos móviles", ha sido aceptado para **participación virtual** en la conferencia.

La versión final de su artículo deberá ser elaborada siguiendo las indicaciones y especificaciones contenidas en la Guía de Autores y remitida a nosotros antes de su fecha tope para el envío de la misma: **12 de enero de 2024**. Usted puede acceder a la Guía de Autores haciendo clic sobre la opción "Autores" que se encuentra en el menú principal del website de la conferencia (<https://www.iiis-spring24.org/cicic>). La misma también contiene una descripción de los pasos requeridos para asegurar la inclusión de su artículo en el Programa de la Conferencia y en las Memorias respectivas.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

En relación a la versión final quisiéramos hacerle dos alertas:

- A partir de este año se requiere que en la versión final del artículo aparezcan el título del mismo y el resumen en inglés. Incluya ambos según las especificaciones de la Guía de Autores. No use otro formato.
- Dada que la calidad de las figuras y, en menor grado, las tablas, es una de las principales causas por la que las versiones finales son devueltas por el *Publisher*, le agradeceríamos que preste atención a lo indicado en la Guía de Autores al respecto: "Estas deberán ser legibles y comprensibles en la versión impresa de las Memorias. Si es necesario para ello, aumente su tamaño o insértelas a lo ancho de las dos columnas, manteniendo siempre, por supuesto, el formato de dos columnas para todo el resto del documento." O, simplemente, no las incluya.

Si la versión final recibida es devuelta por el *Publisher* porque no ha sido preparada siguiendo todas las especificaciones de la Guía de Autores, dependiendo de los tiempos, usted corre el riesgo de que su artículo ya no se pueda incluir en las Memorias que estarán disponibles para los días de la Conferencia.

Recuerde que la responsabilidad de la edición y formato del artículo es única y exclusivamente de los autores.

Cada artículo a ser presentado en el evento podrá ser leído y comentado, vía foros electrónicos, por aquellos participantes que se hayan inscrito en la conferencia.

Los autores que participarán virtualmente en la conferencia tienen la opción (ello no es obligatorio) de:

- enviar un slide show y/o un video pre-grabado de su presentación (presentación asincrónica) or
- hacer su presentación vía Zoom (presentación sincrónica).

Un link para enviar la presentación y/o programar la presentación vía Zoom ha sido provisto en el website de la conferencia bajo la pestaña "Autores".

La inscripción en la conferencia de uno de los autores del artículo, por lo menos, y el pago de las cuotas de registro respectivas son condiciones necesarias para que el artículo sea incluido en las Memorias y/o en el Programa de la Conferencia (virtual); sin embargo, para que su artículo quede publicado en las Memorias que estarán disponibles para los días de la Conferencia, además de inscribirse, se requiere que el pago de sus cuotas de registro sea recibido a más tardar el 29 de febrero de 2024.

Cada cuota de inscripción permite la presentación y/o publicación, en las Memorias de la Conferencia, de un artículo. El autor podrá registrar a lo sumo un artículo adicional por un cargo extra. Ambos artículos deberán ser presentados por el autor inscrito y ser de su autoría.

El número máximo de páginas por artículo es ocho. A lo sumo podrán incluirse dos páginas adicionales más, cancelando la correspondiente cuota por páginas extra.

Para acceder a las evaluaciones y comentarios de los revisores que recomendaron la aceptación de su artículo y enviar la versión final de éste, por favor, visite el website de la conferencia (<https://www.iis-spring24.org/cicic>) y seleccione la opción "Autores" en el menú principal. Usted necesitará ingresar el Código de Identificación de su artículo y la Clave asociada. Para el caso del artículo mencionado en este e-mail de aceptación, éstos son:

Código de Identificación del Artículo: CB701YQ
Clave: CB0405

Adicionalmente le será requerido su Código de Registro en la conferencia. Para obtenerlo, seleccione



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

la opción "Inscripción" en el menú principal del website de la conferencia. Una vez concluido su proceso de registro en línea, Ud. obtendrá una página titulada: "CICIC 2024 - Confirmación de Recepción de Registro". En esta página se le suministrará su Código de Registro. Simultáneamente, se le generará, en forma automática, un e-mail en el cual también se le indicará dicho código.

Si usted, o alguno de sus co-autores, recibe otro e-mail de aceptación de este mismo artículo; pero, con un Código de Identificación y Clave diferentes, ello es debido a que dicho artículo fue propuesto por usted(es) en diferentes oportunidades. En este caso, por favor, envíenos un e-mail a la siguiente dirección: cicic-sec@mail.iiis-spring24.org, indicándonos cuál de los diferentes Códigos de Identificación del Artículo, usted empleará.

Aprovechamos la oportunidad para recordarle que los artículos escritos en cualquier otro idioma distinto a los idiomas oficiales de la Conferencia (español y portugués) no serán publicados en las Memorias ni incluidos en el Programa de la misma.

Esperando contar con su participación en CICIC 2024, le saluda y se despide de Ud.

Muy atentamente,

Prof. Nagib Callaos
Presidente

NOTA: Por favor, no envíe correos electrónicos a esta dirección. La misma es una cuenta del sistema automatizado, por lo que su buzón no es revisado. Para cualquier consulta o asistencia contáctenos a través de la siguiente dirección: cicic-sec@mail.iiis-spring24.org

 **Plense si es necesario imprimir este correo.**
Todos somos responsables por el cuidado del medio ambiente



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019

Anexo II



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



UNLaM - SECyT

FPI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Código: C2-ING-095

Título del Proyecto: Navegación Visual Autónoma por Celular

Director del Proyecto: ~~Mag. Ing. Jorge Eterovic~~

Programa de acreditación: PROINCE

Fecha de inicio: 1/1/2022

Fecha de finalización: 31/12/2023

1. Datos del alumno

Apellido y Nombre: Baltazar Pernas, Agustín

DNI: 42.301.787

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ing. informática

Período evaluado: 2023

2. Dictamen de evaluación de desempeño del alumno:

Colocar una cruz donde corresponda

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno participó en el desarrollo y la implementación exitosa de una prueba de concepto de la participación de un sistema de Visual SLAM, en concreto ~~stella_vslam~~, en una etapa de captura y ~~preprocesamiento~~ de imagen en celular, y ~~posprocesamiento~~ en PC, vía WIFI.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:

3.2 No continuar en el presente proyecto: X

Fundamentos del dictamen:

Su participación en el proyecto ha concluido satisfactoriamente, alcanzando los resultados esperados en diciembre de 2023.

... San Justo, 31/12/23

Lugar y fecha


Firma del Director

Jorge Eterovic
Aclaración de firma



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



UNLaM - SECyT

FPI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas
Código: C2-ING-095
Título del Proyecto: Navegación Visual Autónoma por Celular
Director del Proyecto: Mag. Ing. Jorge Eterovic
Programa de acreditación: PROINCE
Fecha de inicio: 1/1/2022.
Fecha de finalización: 31/12/2023.

1. Datos del alumno

Apellido y Nombre: Bustamante, Juan Andrés
DNI: 39.336.846

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas
Carrera que cursa: Ing. informática
Período evaluado: 2023

2. Dictamen de evaluación de desempeño del alumno:

Colocar una cruz donde corresponda

2.1 Satisfactorio: X
2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno participó en el desarrollo y la implementación exitosa de una prueba de concepto de la participación de un sistema de Visual SLAM, en concreto stella_vslam, en una etapa de captura y preprocesamiento de imagen en celular, y posprocesamiento en PC, vía WIFI.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:
3.2 No continuar en el presente proyecto: X

Fundamentos del dictamen:

Su participación en el proyecto ha concluido satisfactoriamente, alcanzando los resultados esperados en diciembre de 2023.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



UNLaM - SECyT

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

FPI-013

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Código: C2-ING-095

Título del Proyecto: Navegación Visual Autónoma por Celular

Director del Proyecto: ~~Mag. Ing. Jorge Eterovic~~

Programa de acreditación: PROINCE

Fecha de inicio: 1/1/2022

Fecha de finalización: 31/12/2023

1. Datos del alumno

Apellido y Nombre: De Stefano, Matías

DNI: 40.130.248

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ing. informática

Período evaluado: 2022-2023

2. Dictamen de evaluación de desempeño del alumno:

Colocar una cruz donde corresponda

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM, en concreto ~~stella_vs/am~~ en una etapa de captura y preprocesamiento de imagen en celular, y ~~posprocesamiento~~ en PC, vía WIFI.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:

3.2 No continuar en el presente proyecto: X

Fundamentos del dictamen:

Su participación en el proyecto ha concluido satisfactoriamente, alcanzando los resultados esperados en Diciembre de 2023



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



UNLaM - SECyT

FPI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Código: C2-ING-095

Título del Proyecto: Navegación Visual Autónoma por Celular

Director del Proyecto: Mag. Ing. Jorge Eterovic

Programa de acreditación: PROINCE

Fecha de inicio: 1/1/2022.

Fecha de finalización: 31/12/2023.

1. Datos del alumno

Apellido y Nombre: Santos Ferreiro, Leandro Manuel

DNI: 40.496.983

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ing. informática

Período evaluado: 2022-2023

2. Dictamen de evaluación de desempeño del alumno:

Colocar una cruz donde corresponda

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM, en concreto stella_vslam, en una etapa de captura y preprocesamiento de imagen en celular, y posprocesamiento en PC, vía WIFI.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	6
Vigencia	03/9/2019



UNLaM - SECyT

FPI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Código: C2-ING-095

Título del Proyecto: Navegación Visual Autónoma por Celular

Director del Proyecto: Mag. Ing. Jorge Eterovic

Programa de acreditación: PROINCE

Fecha de inicio: 1/1/2022.

Fecha de finalización: 31/12/2023.

1. Datos del alumno

Apellido y Nombre: Sinopoli Alesio, Esteban Abel

DNI: 40.808.298

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ing. informática

Período evaluado: 2022-2023

2. Dictamen de evaluación de desempeño del alumno:

Colocar una cruz donde corresponda

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno participó en el desarrollo y la implementación exitosa de una prueba de concepto de la partición de un sistema de Visual SLAM, en concreto *stella_vslam*, en una etapa de captura y preprocesamiento de imagen en celular, y posprocesamiento en PC, vía WIFI.