TLÁLOC-I, satélite enlatado con modelo de machine learning para detección de agua.

TLÁLOC-I, canned satellite with machine learning model for water detection.

Octavio Mendoza Gómez[1]; Luis Fernando Ángel Vilchis Chalate[2]; Óliver Nájera Reyes[3]
[1,2,3]Universidad Popular Autónoma del Estado de Puebla, México, *octavio.mendoza@upaep.edu.mx, luisfernando.vilchis@upaep.edu.mx, oliver.najera@upaep.mx*

The development of the CanSat called Tláloc-I, focuses on the ability to collect hydrological data from remote areas from an altitude of three kilometers using image capture by camera for subsequent analysis with a machine learning model, this arises due to the lack of water sources to continue with daily activities such as agriculture and livestock between others; this resource is important and there are areas where humans can not easily reach. The target altitude will be reached by a launcher with gyroscopes and parachutes, ensuring a safe landing for the payload. The CanSat is equipped with multiple sensors whose data is processed by a Python program, stored in-text files and sent via serial communication to the operator location of the human machine interface. Real-time monitoring is performed with LabVIEW as the receiving interface of data from the serial port of XBee. The essential functions of the program have been successfully implemented and the team is currently working underway on the manufacture of printed circuit boards for electronic systems. The conclusions focus on the lessons learned during the development process with the expectation of successfully achieving the mission of collecting hydrological data in remote areas. It is relevant to highlight the increasingly important role of new technologies such as artificial intelligence, neural networks as well as learning models, which are transforming many aspects of daily life, including the economic one. This project contributes to improving the lives of farmers, ranchers and residents in areas with water scarcity.

Key words: Machine Learning, Artificial Intelligence, Canned Satellite, Human Machine Interface, Vehicle Launcher.

TLÁLOC-I, satélite enlatado con modelo de machine learning para detección de agua.

Octavio Mendoza Gómez[1]; Luis Fernando Ángel Vilchis Chalate[2]; Óliver Nájera Reyes[3]
[1,2,3]Universidad Popular Autónoma del Estado de Puebla, México, *octavio.mendoza@upaep.edu.mx, luisfernando.vilchis@upaep.edu.mx, oliver.najera@upaep.mx*

El desarrollo del CanSat llamado Tláloc-I se centra en la capacidad de recolectar datos hidrológicos de áreas remotas desde una altitud de tres kilómetros usan capturas de imagen de una cámara para su posterior análisis con un modelo de machine learning, esto surge debido a la falta de fuentes de agua que permitan las actividades diarias como la agricultura y la ganadería entre otras, pues este recurso es importante y existen áreas donde los humanos no pueden acceder a él fácilmente. La altitud objetivo será alcanzada mediante un lanzador con giroscopios y paracaídas asegurando un aterrizaje seguro de la carga útil. El CanSat está equipado con múltiples sensores cuya información es procesada mediante un programa de Python, almacenado en archivos de texto y enviada mediante comunicación serial a la ubicación del operador de la interfaz humano-máquina. El monitoreo en tiempo real se realiza con LabVIEW que actúa como la interfaz receptora de los datos provenientes del puerto serial de XBee. Las funciones esenciales del programa se han implementado con éxito y el equipo está trabajando actualmente para la fabricación de circuitos impresos del sistema electrónico. La conclusión se enfoca en las lecciones aprendidas durante el desarrollo con la expectativa de lograr éxito en la misión de recolectar datos hidrológicos en áreas remotas. Es relevante destacar el importante papel de las nuevas tecnologías como la inteligencia artificial, redes neuronales y modelos de aprendizaje que están transformando muchos aspectos de la vida diaria, incluyendo el aspecto comercial. Este proyecto contribuye a mejorar la vida de los agricultores, ganaderos y residentes en zonas con escasez de agua.

Palabras clave—machine learning, inteligencia artificial, satélite enlatado, interfaz humano-máquina, vehículo lanzador.

I. Introducción

Los recursos naturales del planeta son fundamentales para satisfacer las necesidades básicas de la humanidad. Sin embargo, el uso desmedido en conjunto con prácticas ineficientes para su aprovechamiento ha llevado a una creciente preocupación por su escasez. Este problema ha impulsado el desarrollo de técnicas para mejorar la eficiencia en la extracción y uso de estos recursos, especialmente en regiones donde la disponibilidad es crítica.

El agua, uno de los recursos naturales más esenciales para la vida, es un claro ejemplo de esta problemática. Su disponibilidad ha disminuido significativamente en muchas áreas de México, lo que ha llevado a la necesidad de restringir su consumo preventivamente. En algunos lugares el acceso al agua es limitado intermitentemente debido a su escasez. Según la Organización de las Naciones Unidas (ONU), aproximadamente 2 mil millones de personas o lo que puede traducirse al 26% de la población mundial, carecen de un acceso seguro al agua potable. Estas personas viven en áreas que enfrentan graves problemas de escasez de agua, lo que afecta negativamente su calidad de vida obligándoles a arriesgarse a recorrer grandes distancias para obtener este recurso vital.

Esta situación no solo impacta a las comunidades afectadas, sino que también influye en la calidad de vida de regiones enteras, dado que la gestión de los recursos hídricos es crucial para el bienestar y el desarrollo de muchos entornos. Según datos del Instituto Nacional de Estadística y Geografía (INEGI), el uso eficiente del agua es crucial no solo para la base de la alimentación mexicana, sino también para el desarrollo económico de ganaderos, agricultores y empresas que generan un gran número de empleos estables. El agua es esencial en la vida diaria, desde la higiene y la limpieza, hasta la alimentación.

Con el objetivo de mejorar el aprovechamiento de los recursos hídricos en el país, nace Tláloc-I, un proyecto que busca soluciones innovadoras. Tláloc-I es un CanSat diseñado para el rastreo hidrológico el cual se elevará a 3 kilómetros de altitud con el propósito de tomar fotografías de áreas específicas, identificando así la presencia de agua en grandes entornos geográficos. De la mano de herramientas como el machine learning, un procesamiento de imágenes aéreas permitirá detectar zonas con flujos constantes de agua, facilitando su aprovechamiento para las comunidades circundantes. Este proyecto tiene como objetivo recopilar información detallada sobre los recursos hídricos, lo que permitirá una planificación estratégica de su uso, optimizando el acceso y el tiempo de explotación. [1]

II. Metodología

El desarrollo de Tlaloc-1, abarca desde el uso de sensores para monitorear el estado del CanSat hasta la programación de un sistema que permita una futura recolección y procesamiento de imágenes hidrológicas mediante un modelo de Machine Learning. Para posteriormente concluir con la correcta comunicación en tiempo real entre el satélite enlatado y una estación terrena situada a una distancia segura.

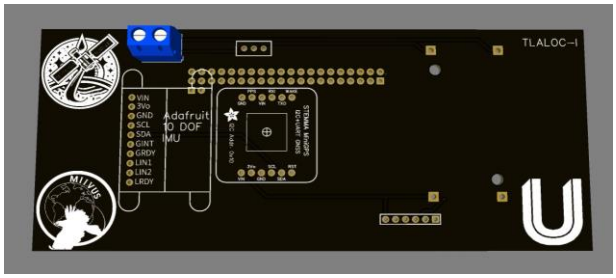

Figura 1. Vista frontal de la PCB

B. Diseño y Hardware

En el diseño de un CanSat se busca el aprovechamiento de un espacio reducido similar al de una lata donde se monitoree información mediante sensores y su comunicación con un microcontrolador. En este caso se utiliza una Raspberry Pi Zero W (RPi Zero) conectada a los siguientes sensores: un GPS (Adafruit Mini GPS PA1010D), una unidad de medición inercial (Adafruit 10-DOF IMU), una antena Xbee Pro, un convertidor DC-DC y una cámara.

El GPS indicará la posición en coordenadas en tiempo real para monitoreo y recuperación del dispositivo al caer, el sensor de medición inercial es capaz de medir velocidad, aceleración y fuerza gravitacional en el dispositivo para obtener de manera precisa su orientación, además, incluye un barómetro que ayuda a determinar la altitud del satélite mediante la variación de presión, la antena permite la recopilación y envío de información de la tarjeta a la estación terrena mediante comunicación serial en sus puertos, el convertidor de corriente continua (DC-DC) permite la manipulación del voltaje necesario para la alimentación del circuito y que este permanezca estable, por último, la cámara es la responsable de tomar fotografías para enviarlas al almacenamiento de la tarjeta para su posterior análisis.

La alimentación de los sensores se realiza con pilas de litio recargables de 7.4 volts (V) conectadas a un regulador para disminuir su voltaje y así asegurar que el voltaje demandado por el circuito será el adecuado para su funcionamiento. El consumo nominal total es de 5 V y 600 miliamperios (mA). El diagrama electrónico de la tarjeta de vuelo (Fig. 1) del satélite enlatado se realizó para su impresión en placa de circuito impreso (PCB) y los dispositivos externos serán conectados con pines hembra soldados a la placa. [2][3][4][5]

C. Software

El procesamiento de información está basado en el lenguaje de programación CircuitPython, donde se realiza la inicialización de sensores y lectura mediante librerías diseñadas específicamente para el aprovechamiento del hardware. La comunicación utilizada entre los sensores y la Raspberry Zero es mediante protocolo I2C, donde con un solo bus de datos se coordina el envío de información a una frecuencia de 100kHz, las variables se inician en cero y el programa principal consiste en un bloque 'while True' infinito. En este se toman lecturas cada 250 milisegundos donde se envía la información recibida mediante comunicación serial a la antena XBee. La idea de la lectura de los sensores es recabar información en tiempo real del entorno en el que se encuentra volando el dispositivo, pero el de mayor importancia para el control de vuelo es el barómetro, cuyo valor de presión indicará la altitud a la que se encuentra: una vez encendida toda la electrónica, el sensor hace su primera medición y la guarda como referencia, posteriormente sigue tomando mediciones en cada ciclo y obtiene la diferencia del valor de referencia, donde dependiendo del cambio de presión este indicará si el cohete se elevó o si descendió, posteriormente se guarda un valor extra cada 3 ciclos cuyo valor se compara con el valor más reciente, de esta manera el cohete detecta si está subiendo o bajando y esto permite activar la ignición para la apertura del paracaídas. Los datos enviados a la estación terrena se envían en una frecuencia de 2.4 GHz. [6]

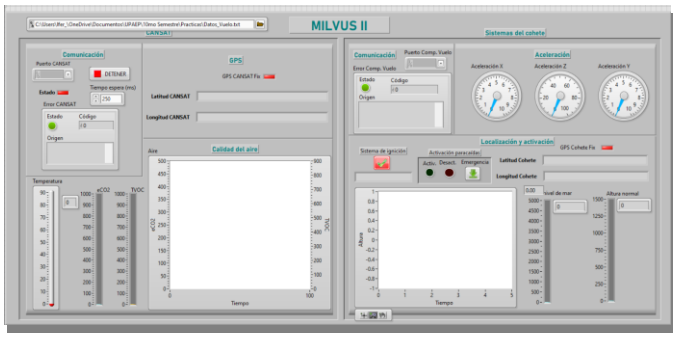

Fig. 2 Interfaz humano máquina elaborada en LabVIEW

D. Estación terrena

La estación terrena es la encargada de recibir los datos enviados desde el satélite, en este caso se usa una antena yagi de 15 elementos de alta ganancia de 12.5 decibeles (dBi) (Digi, 2024, p218). Esta antena se comunica con el módulo XBee, y envía la información a la computadora por medio de un cable USB tipo A, en protocolo serial. La recepción de datos se hace mediante una interfaz humano máquina, elaborada en Labview con los bloques VISA, donde se recibe la información en variables tipo 'string' y posteriormente se manipulan dependiendo de su posición para extraer el valor necesario y presentarlo en el 'Front Panel' con métodos gráficos o numéricos. También cuenta con indicadores y botones booleanos para mandar y recibir confirmaciones de estado de los sistemas, ignición y funcionamiento del GPS. [7][8][9]

E. Machine Learning

El lenguaje de programación que se usa es Python, esto, debido a que se cuentan con librerías robustas para el análisis de datos, así como para el modelo de aprendizaje.
Las librerías utilizadas son Numpy, Pandas, GDAL (para el manejo de datos geoespaciales), scikit-learn (para el aprendizaje automático) y matplotlib (para la graficación).

La preparación de los datos implica abrir y leer archivos de imágenes satelitales GeoTIFF (datos de dominio público de información georreferenciada) mediante

GDAL. El script extrae datos de entrenamiento y prueba de estos archivos y los convierte en Pandas DataFrames para su posterior procesamiento.

Para la visualización de los datos de imágenes satelitales y sus bandas espectrales mediante EarthPy y matplotlib. Esto ayuda a comprender las características y la distribución de los valores de los píxeles en diferentes bandas.

El parámetro que se ocupa para la detección de agua en imágenes es el Índice de Vegetación de Diferencia Normalizada (NDVI) el cuál es usado habitualmente para cuantificar la densidad y la salud de la vegetación.

$$NDVI = \frac{NIR - rojo}{NIR + rojo} \quad (1)$$

Donde:
NIR: valores de píxeles de la banda de infrarrojo más cercano.
rojo: valores de píxeles de la banda roja. [10]

Una vez obtenido los datos, se procede al desarrollo del modelo de machine learning usando métodos de clasificación como k-n vecinos y random forest. El algoritmo k-Nearest Neighbors (k-NN) es un método simple e intuitivo para tareas de clasificación. Es un tipo de aprendizaje basado en instancias o aprendizaje perezoso en el que el modelo aprende directamente de los datos de entrenamiento. El principio detrás de k-NN es predecir la clase de un nuevo punto de datos en función de la clase mayoritaria entre sus k vecinos más cercanos en el espacio de características.

El objetivo del algoritmo del vecino más cercano es identificar los vecinos más cercanos de un punto de consulta dado, será necesario calcular la distancia entre el punto de consulta y los otros puntos de datos. Estas métricas de distancia ayudan a formar límites de decisión, que dividen los puntos de consulta en diferentes regiones. Hay varias medidas de distancia, se usaron euclidiana, esta es la medida de distancia más utilizada y está limitada a vectores de valor real; manhattan, esta es también otra métrica de distancia popular, que mide el valor absoluto entre dos puntos. También se conoce como distancia de taxi o distancia de cuadra de la ciudad, ya que comúnmente se visualiza con una cuadrícula que ilustra cómo se puede navegar de una dirección a otra a través de las calles de la ciudad y minkowski, medida de distancia la cuál es la forma generalizada de las métricas de distancia Euclidiana y Manhattan.

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (yi - xi)^2} \quad (2)$$

El parámetro, p=2, en (2) permite la creación de otras métricas de distancia. La distancia euclidiana se representa mediante esta fórmula cuando p es igual a dos, calcula la distancia entre dos puntos.

$$d(x,y) = \sqrt{\sum_{i=1}^{m} |xi - yi|} \quad (3)$$

La distancia de Manhattan se denota con p igual a uno, la fórmula ocupa el valor absoluto entre dos puntos como en la fórmula (3). Este algoritmo será utilizado para realizar la predicción de datos por medio de la comparación de un punto, denominado k, con puntos en su cercanía. [11][12]

$$distancia\ Minkowski = \left(\sum_{i=1}^{n} |xi - yi|\right)^{1/p} \quad (4)$$

La fórmula de distancia minkowski (4) es la forma generalizada de las métricas euclidiana y manhattan, a partir de esta, se crean otras métricas de distancia.

Random forest o (“el bosque aleatorio” en español) es un método de aprendizaje conjunto que construye múltiples árboles de decisión durante el entrenamiento y genera como resultado la moda de las clases (clasificación) o la predicción media (regresión) de árboles individuales. Para el modelo de random forest, de igual manera se realiza búsqueda en cuadrícula. Es necesario ajustar los hiper parámetros del algoritmo de bosque aleatorio para identificar la mejor combinación para lograr un rendimiento óptimo del modelo. La búsqueda en cuadrícula ayuda a buscar sistemáticamente en una cuadrícula de parámetros específica y a seleccionar los mejores parámetros en función del rendimiento validado de forma cruzada. Ambos son modelos de aprendizaje ocupados para lograr la misión principal del satélite enlatado, los modelos han sido probados y han funcionado correctamente. [13]

III. Resultados

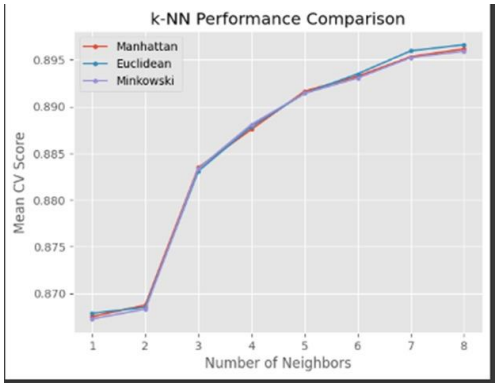

Fig. 3 Gráfico de performance para k-n vecinos

En la (Fig. 3), se muestra el gráfico de comparación de performance entre los tres tipos de distancia, euclidiana, manhattan y minkowski, dónde el gráfico de los tres parámetros, son muy similares entre sí.

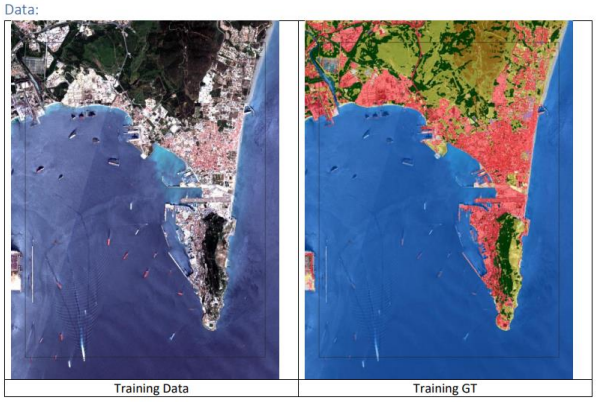

Fig. 4 Aplicación del algoritmo en imagen de prueba.

En la (Fig. 4) se visualiza el éxito en la detección de agua, usando el algoritmo en la imagen de entrenamiento, incluso se hace la detección de edificios y zonas verdes, marcados de color rojo y verde respectivamente.

IV. Conclusiones

El desarrollo del proyecto ha sido óptimo y funcional, incluyendo el diseño electrónico, la programación, la telemetría y la misión principal que es el desarrollo del modelo de machine learning. Sin embargo, consideramos que, para todo proyecto e investigación, es primordial seguir avanzando en el desarrollo. Es por ello que realizar pruebas en vuelo con vehículo lanzador serán relevantes para el avance del mismo.

La realización de este proceso llevado a cabo con una metodología específica nos ha permitido avanzar y mejorar el desarrollo del mismo. Esto, en adición a las nuevas tecnologías, en específico el machine learning, permite el uso de tecnologías de vanguardia para la realización de proyectos para el logro de los objetivos. Esto se puede ver reflejado en los resultados del proyecto, demostrando avances en la optimización del algoritmo como en el correcto funcionamiento de los programas encargados de detectar el agua.

Agradecimiento/Reconocimiento

Agradecemos el apoyo de la facultad de Mecatrónica, Biónica y Aeroespacial de la UPAEP; de los directivos, Dr. Omar Gónzalez Gónzalez, Dra. Argelia Fabiola Miranda Pérez, por el excelente seguimiento que brindan; del asesor del proyecto de cohetería experimental Milvus Rocket de la UPAEP, Mtro. Luis Felipe Córdova Estrada; al igual que a todos los integrantes del equipo de cohetería experimental. De igual manera el apoyo de la Dirección de Investigación de la UPAEP, Lic. Bryan Martínez Rodríguez, Dra. Vanessa Sedeño Monreal Vera, Mtra. Johanna Olmos López y Dr. Manuel Ponce de León por inculcar la importancia de la vocación científica y de investigación en las juventudes. Agradecer a nuestras familias por ayudarnos a conseguir nuestras metas y sueños en nuestro recorrido académico y profesional.

References

[1] Sinforoso Martínez, S., Pelegrín Mesa, A., & Álvarez Velázquez, E. (2020). Contribución del costo sostenible para el cuidado del agua. Una mirada desde la Agenda 2030. *Retos de la Dirección*, *14*(2), 205-224.

[2] Islam, T., Noureen, A., Mughal, M. R., & Nadeem, M. A. (2019, December). Design and development of a weather monitoring satellite, CanSat. In *2019 15th International Conference on Emerging Technologies (ICET)* (pp. 1-6). IEEE.

[3] Lady Ada (2017). Adafruit LSM9DS1 Accelerometer + Gyro + Magnetometer 9-DOF Breakout. https://learn.adafruit.com/adafruit-lsm9ds1-accelerometer-plus-gyro-plus-magnetometer-9-dof-breakout/overview

[4] Bosch (2013) BMP180, Digital pressure sensor. *https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Pressure/BMP180.pdf*

[5] DIGI (2022).XBee®/XBee-PRO S2C Zigbee. *https://www.digi.com/resources/documentation/digidocs/pdfs/90002002.pdf*

[6] MicroPython & CircuitPython contributors. (2025). CircuitPython. https://docs.circuitpython.org/en/latest/README.html

[7] Ponce-Cruz, P., & Ramírez-Figueroa, F. D. (2010). *Intelligent control for labview* (pp. 1-8). Springer London.

[8] National Instruments (2022). Building LabVIEW User Interfaces https://www.ni.com/en/support/documentation/supplemental/08/building-labview-user-interfaces.html/1000?srsltid=AfmBOorYw77Za7X_6N58-_UUqTPxUiuhKUHVLl4UNzixU4nNA3lTQA-x

[9] Analog Devices, (2002). 2-Wire Communication Using LabVIEW. https://www.analog.com/en/resources/technical-articles/2wire-communication-using-labview.html

[10] Meneses-Tovar, C. L. (2011). NDVI as indicator of degradation. *Unasylva*, *62*(238), 39-46.

[11] Magidi, J., Nhamo, L., Mpandeli, S., & Mabhaudhi, T. (2021). Application of the Random Forest Classifier to Map Irrigated Areas Using Google Earth Engine. *Remote Sensing*, *13*(5), 876. https://doi.org/10.3390/rs13050876

[12] Kim, J. M., Park, J. S., Lee, C. Y., and Lee, S. G.: PREDICTING OF URBAN EXPANSION USING CONVOLUTIONAL LSTM NETWORK MODEL: THE CASE OF SEOUL METROPOLITAN AREA, KOREA, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., X-4/W3-2022, 113–118, https://doi.org/10.5194/isprs-annals-X-4-W3-2022-113-2022, 2022.

[13] Wadii Boulila, Hamza Ghandorh, Mehshan Ahmed Khan, Fawad Ahmed, Jawad Ahmad, A novel CNN-LSTM-based approach to predict urban expansion, Ecological Informatics, Volume 64, 2021, 101325, ISSN 1574-9541, https://doi.org/10.1016/j.ecoinf.2021.101325. (https://www.sciencedirect.com/science/article/pii/S1574954121001163)J .-G. Lu, "Title of paper with only the first word capitalized," *J. Name Stand. Abbrev.*, in press.