



Propuesta de un sistema de monitoreo agrícola basado en técnicas de IA

Angela Pérez Florentino^{1}, Héctor Hugo Moreno Pérez¹ y Argia Lilí Paz Molina¹*

¹TecNM-Instituto Tecnológico Superior de Pánuco

*angela.perez@itspanuco.edu.mx

RESUMEN

El diseño de un prototipo de vehículo autónomo para monitoreo agrícola es muy importante, ya que permite actualizar los conocimientos y aplicarlos a favor de nuestra sociedad, al proponer soluciones tecnológicas como la aplicación de las redes convolucionales o el aprendizaje profundo, que les apoyen en actividades que al realizarlas de forma manual se tardarían mas tiempo en llevarlas a cabo. Se realizó una revisión documental para conocer y elegir las herramientas tecnológicas a utilizar en el vehículo experimental, posteriormente se realizó el armado del prototipo continuando con la recopilación de imágenes, después se hizo el entrenamiento del modelo y se hicieron las pruebas para corroborar que funcione. Así mismo, se presentan los resultados de esta primera fase del proyecto, que muestran el vehículo en funcionamiento al estar capturando las imágenes y procesandolas en el modelo de entrenamiento, permitiendo que el vehículo navegue evitando obstáculos.

Palabras claves: Vehículo Autónomo, Redes Convolucionales, Aprendizaje Profundo.

ABSTRACT

The design of an autonomous vehicle prototype for agricultural monitoring is very important, as it allows us to update knowledge and apply it in favor of our society, by proposing technological solutions such as the application of convolutional networks or deep learning, which support them in activities that, if carried out manually, would take longer to carry out. A documentary review was carried out to know and choose the technological tools to be used in the experimental vehicle, then the prototype was assembled, continuing with the collection of images, then the model was trained, and tests

were carried out to corroborate that it works. Likewise, the results of this first phase of the project are presented, which show the vehicle in operation by capturing the images and processing them in the training model, allowing the vehicle to navigate avoiding obstacles.

Key words: Autonomous Vehicle, Convolutional Networks, Deep Learning

INTRODUCCIÓN

El uso de la tecnología es cada vez más utilizado y sofisticado, atendiendo desde las necesidades más básicas hasta las más complejas. Existen una diversidad de tecnologías y redes neuronales para el aprendizaje autónomo: deep learning y redes convolucionales, cada una con sus espacios o áreas de aplicación.

En la actualidad se evidencia un fuerte crecimiento de la automatización de las labores agrícolas atendiendo a la evolución tecnológica y a la expansión poblacional del planeta. De lo anterior surge un interrogante que obedece a la tendencia mundial de la industria 4.0: ¿Cuáles tecnologías son las más adecuadas y apropiadas para desarrollar proyectos en el sector agrícola? Debido a esto, es importante identificar el estado actual de las aplicaciones agrícolas y sus tecnologías, y establecer cuáles apuntan a los desarrollos y desafíos del nuevo siglo. (Tovar, 2019)

De acuerdo a González (2015) la agricultura en todo el mundo propende por la seguridad alimentaria a futuro, razón por la cual se está en la búsqueda de tecnologías modernas que ayuden a mitigar la falta de atención de ciertas particularidades de los cultivos que generan gastos innecesarios en el monitoreo de los mismos. De igual forma menciona que las extensiones de las fincas no eran tan grandes, por lo que los cultivos tampoco lo eran como en la actualidad, lo que hace más tardado el recorrerlos en su totalidad.

Según Pino (2019), una actividad común en la agricultura es la aplicación de pesticidas o de fertilizantes (nitrógeno, fósforo, potasa) y micronutrientes (azufre, magnesio, zinc). Tanto el fertilizante como pesticidas son aplicados por equipos como bombas rociadoras manuales, pulverizadores de tractor o sistemas de riego a presión.

He aquí la importancia del uso de vehículos autónomos que faciliten los recorridos por las amplias y extensas áreas de cultivo para apoyar en las actividades propias de los mismos. Un vehículo autónomo de acuerdo a Gómez (2023), para ser desarrollado se debe enfocar más en la parte de la visión, por lo que se enfoca en una de las ramas más

estudiadas de la inteligencia artificial: la visión artificial. Por lo que dice que es la encargada de dar a una máquina la capacidad de ver y cita a (Nalwa, 1993), quien dice que: “La visión artificial o comprensión de imágenes describe la deducción autónoma de la estructura y propiedades de un mundo tridimensional, posiblemente dinámico, bien a partir de una o varias imágenes bidimensionales de ese mundo” por lo que la visión, tanto para los humanos como para un sistema, consta de dos fases principalmente: captar una imagen e interpretarla.

Este documento versa sobre el desafío de la navegación autónoma de un vehículo experimental empleando Redes Neuronales Convolucionales (Convolutional Neural Network [CNN]), que de acuerdo a Castillo (2023), son una clase de algoritmos de aprendizaje profundo especialmente diseñados para el procesamiento de imágenes y la detección de patrones.

Con el desarrollo e implementación de este vehículo experimental se explorarán las capacidades de la IA en la toma de decisiones autónomas en la navegación en entornos agrícolas.

El objetivo de este trabajo de investigación es el diseño de un vehículo experimental en su primera fase de prueba, que sea capaz de navegar en forma autónoma al ser entrenado el modelo para que pueda evitar los obstáculos en primera instancia.

METODOLOGÍA

Actualmente el aprendizaje automático o de máquina esta siendo muy utilizado, Silva (2020) lo describe como una rama de la inteligencia artificial que tiene como objetivo, desarrollar algoritmos computacionales que doten la capacidad de “aprender” a las computadoras.

Para el diseño del vehículo experimental se utiliza el método de investigación experimental porque se trabaja directamente con el primer prototipo del vehículo autónomo, mismo que hasta el momento evita obstáculos. En cuanto al diseño de la investigación es pre-experimental porque se estará observando el comportamiento en la aplicación del vehículo prototipo.

Como primer paso para el diseño del prototipo, se ha hecho una revisión documental para conocer las tecnologías y redes neuronales para el aprendizaje autónomo, en donde se

encontró que se puede trabajar con el aprendizaje profundo (deep learning) o redes neuronales convolucionales (CNN), inclinándose por este último.

Lo anterior porque de acuerdo a AWS (s.f), una red neuronal es la tecnología subyacente en el aprendizaje profundo, la cual consiste en nodos o neuronas interconectados en una estructura en capas. Estos nodos procesan los datos en un sistema coordinado y adaptativo, los cuales intercambian comentarios sobre los resultados generados, aprenden de los errores y mejoran continuamente. Por lo tanto, las redes neuronales artificiales son el núcleo de un sistema de aprendizaje profundo.

Es así que se seleccionó a las redes neuronales convolucionales porque son capaces de extraer el modelo ajustando parámetros como pesos de la red, para minimizar el error entre las predicciones del modelo y las etiquetas de control reales como la dirección y la velocidad.

Como segundo paso, se analizaron que componentes utilizar para este primer prototipo de prueba, decidiéndose por los que se enlistan a continuación:

1. Raspberry Pi. Se utiliza para controlar la locomoción, la adquisición de las imágenes y la operación del modelo de inteligencia artificial.
2. Driver LN298. Usado para controlar los motores.
3. Servo Motor. Para controlar la tracción delantera del vehículo prototipo.
4. Motores DC. Sirve para controlar el Driver LN298.
5. Kit carrito. Utilizado para montar todos los componentes.
6. Cámara Raspberry Pi. Destinada a la adquisición de imágenes que entrenaran el modelo.
7. Computadora con más de 3GHZ de procesamiento y 16GB de RAM.

Como tercer paso, se armó el vehículo prototipo y se procedió a la recopilación de las imágenes para posteriormente procesarlas con el modelo de entrenamiento y realizar las pruebas correspondientes para verificar su funcionamiento.

RESULTADOS

A continuación se describen los resultados del procedimiento realizado en el diseño del prototipo.

Se trabajó sobre el sistema operativo ROS (Sistema Operativo para Robots), que de acuerdo a (ROS, s.f.), es un conjunto de bibliotecas de software y herramientas que ayudan a crear aplicaciones para robots. De igual forma se utilizó la librería Donkeycar (imagen 1) que permite el desplazamiento del vehículo experimental.

```
#Define a vehicle to take and record pictures 10 times per second.

import time
from donkeycar import Vehicle
from donkeycar.parts.cv import CvCam
from donkeycar.parts.tub_v2 import TubWriter
V = Vehicle()

IMAGE_W = 160
IMAGE_H = 120
IMAGE_DEPTH = 3

#Add a camera part
cam = CvCam(image_w=IMAGE_W, image_h=IMAGE_H, image_d=IMAGE_DEPTH)
V.add(cam, outputs=['image'], threaded=True)

#add tub part to record images
tub = TubWriter(path='./dat', inputs=['image'], types=['image_array'])
V.add(tub, inputs=['image'], outputs=['num_records'])

#start the drive loop at 10 Hz
V.start(rate_hz=10)
```

Imagen 1. Librería Donkeycar.

En la imagen 2, se muestra el vehículo experimental armado con su computadora a bordo.

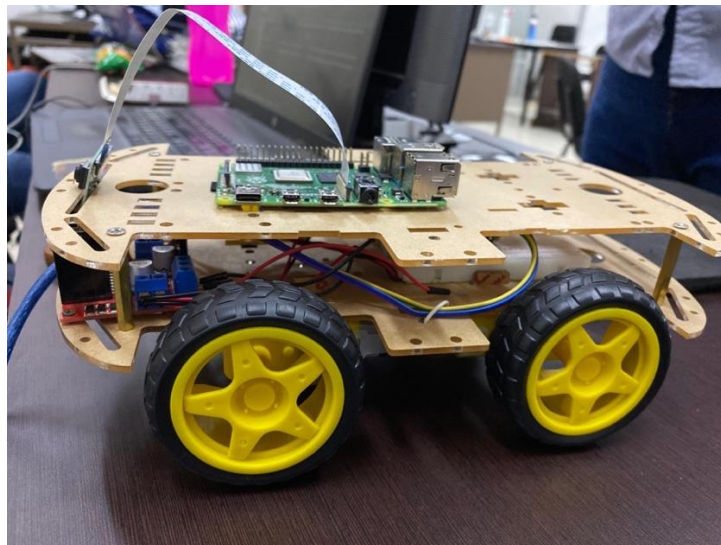


Imagen 2. Vehículo prototipo.

Para entrenar el modelo se tienen que capturar las imágenes a una frecuencia de entre 10 a 20 imágenes por segundo (imagen 3), mismas que el framework Donkeycar registra con etiquetas tomando en cuenta la dirección, velocidad y la posición de la cámara; lo

que permite al prototipo del vehículo autónomo recorrer la ruta deseada para ir registrando las imágenes.

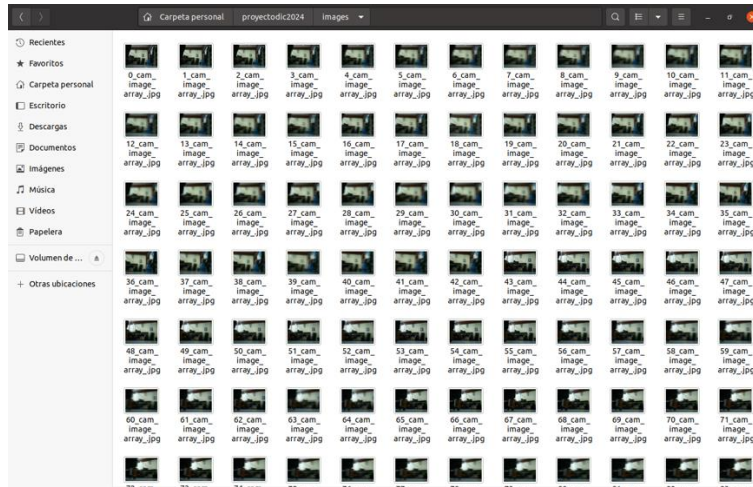


Imagen 3. Dataset obtenido desde el prototipo.

Posteriormente Donkeycar las redimensiona a un tamaño mas pequeño para acelerar el proceso de entrenamiento (imagen 4), lo cual hace utilizando la librería Keras que se encarga de extraer las características más importantes como bordes, formas y patrones en las imágenes, que permitirá al vehículo prototipo decidir a que dirección ir y a que velocidad.



Imagen 4. Imágenes varias capturadas (editadas por software Pica AI porque Donkeycar ya las había redimensionado y etiquetado).

La siguiente etapa es la normalización de datos, que consiste en transformar sus valores de píxeles y se ajustan para estar dentro de un rango específico (por ejemplo, entre 0 y 1), lo que ayuda al modelo a aprender de manera más eficiente.

Arquitectura de la red neuronal (imagen 5):

1. Entradas:

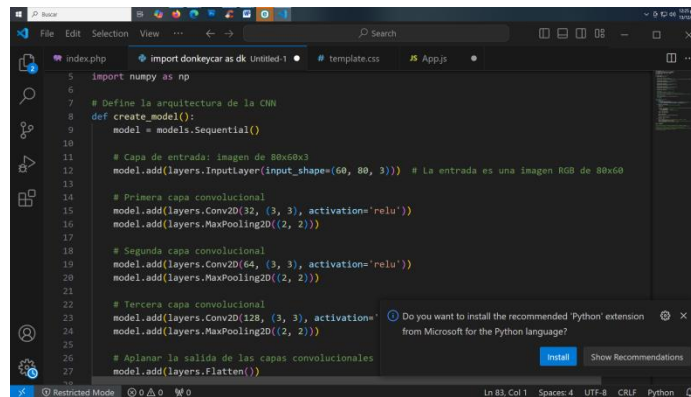
- Imágenes de tamaño 160x120x3 (alto, ancho, canales de color).
- Capas convolucionales: 3 capas convolucionales con 24, 36 y 48 filtros de tamaño 5x5.
- Capas de Max Pooling: 3 capas con tamaño de pool 2x2.
- Capas completamente conectadas: 4 capas densas con 1164, 100, 50 y 10 unidades, respectivamente.
- Salida: 2 valores continuos (dirección y velocidad).

2. Parámetros de Entrenamiento:

- Función de pérdida: Error cuadrático medio (MSE).
- Optimizador: Adam.
- Épocas: Dependiendo de la cantidad de datos y el tiempo de entrenamiento, el modelo puede entrenarse durante varias épocas (por ejemplo, 10-50).

3. Salida del modelo: Generalmente tiene dos salidas principales:

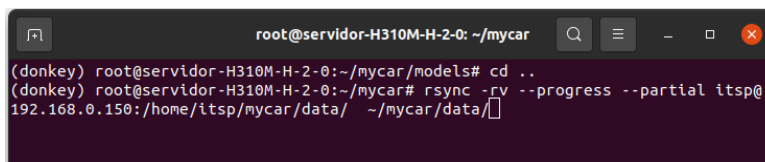
- Dirección: Qué tanto debe girar el volante.
- Velocidad: Cuánto debe acelerar el coche.



```
5 import numpy as np
6
7 # Define la arquitectura de la CNN
8 def create_model():
9     model = models.Sequential()
10
11     # Capa de entrada: imagen de 80x80x3
12     model.add(layers.InputLayer(input_shape=(60, 80, 3))) # La entrada es una imagen RGB de 80x60
13
14     # Primera capa convolucional
15     model.add(layers.Conv2D(32, (3, 3), activation='relu'))
16     model.add(layers.MaxPooling2D((2, 2)))
17
18     # Segunda capa convolucional
19     model.add(layers.Conv2D(64, (3, 3), activation='relu'))
20     model.add(layers.MaxPooling2D((2, 2)))
21
22     # Tercera capa convolucional
23     model.add(layers.Conv2D(128, (3, 3), activation='relu'))
24     model.add(layers.MaxPooling2D((2, 2)))
25
26     # Aplanar la salida de las capas convolucionales
27     model.add(layers.Flatten())
```

Imagen 5. Descripción de la arquitectura de la red neuronal.

Para la generación del modelo el framework Donkeycar (imagen 6) llama a los dataset de imágenes como tubs.



```
root@servidor-H310M-H-2-0: ~/mycar
(donkey) root@servidor-H310M-H-2-0:~/mycar/models# cd ..
(donkey) root@servidor-H310M-H-2-0:~/mycar# rsync -rv --progress --partial itsp@192.168.0.150:/home/itsp/mycar/data/ ~/mycar/data/
```

Imagen 6. Proceso de transferencia del dataset de imágenes obtenidas.

En la interfaz del framework se analizan los tubs (imagen 7).

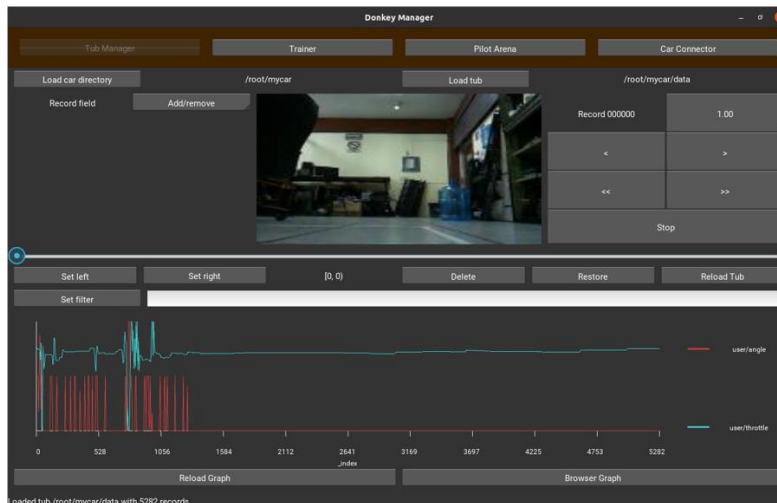


Imagen 7. Dataset con 5280 imágenes.

El entrenamiento se realiza en la opción trainer del framework de la interfaz de usuario (UI) o en la ventana de comandos. (imagen 8)



Imagen 8. Entrenamiento.

Los resultados del entrenamiento se muestran en el framework UI. (imagen 9)

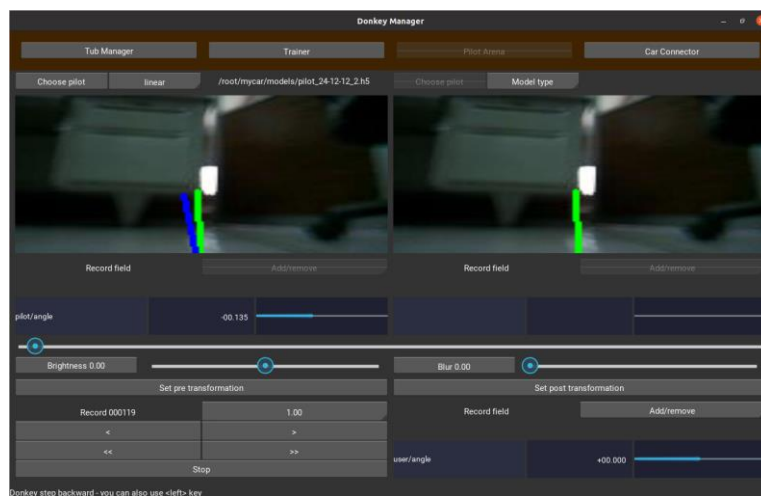


Imagen 9. Resultados del entrenamiento.

La línea verde es el dato capturado en el recorrido y la azul es la del entrenamiento. Solo se llegó hasta esta etapa para crear el modelo y no se analizó la presión del mismo.

DISCUSIÓN

En esta investigación se presenta el diseño de un vehículo autónomo basado en la redes neuronales convolucionales, cuyo objetivo es el de explorar las capacidades de la Inteligencia Artificial en la toma de desiciones autónomas. Es así que en este primer prototipo se esta entrenando al vehículo para que sea capaz de evitar obstáculos y sea capaz de navegar de forma autónoma.

Lo anterior tomando en cuenta trabajos de investigación que usan el aprendizaje automático, cuyas herramientas realizan la recoleccion de información que posteriormente es procesada para permitir la toma de decisiones.

CONCLUSIONES

Con el diseño de este tipo de vehículos de monitoreo se pretende coadyuvar a las personas que trabajen algun tipo de cultivo y que en determinado momento requieran de realizar tareas que son repetitivas, proponiendo este tipo de tecnología autónoma adecuandola de acuerdo a sus necesidades. Cabe hacer mención que este vehículo es una primera prueba y se proyecta mejorarlo tanto en infraestructura como en funcionalidades. Es así como se concluye que al existir una diversidad de vehículos autónomos tanto terrestres como aéreos, siempre queda el espacio para seguir explorando y darle utilidad a estos trabajos o proyectos ya realizados en otros lugares, adaptándolos con los recursos a nuestro alcance a las necesidades de nuestra región.

LITERATURA CITADA

- AWS. s.f. *¿Cuál es la diferencia entre el aprendizaje profundo y las redes neuronales?*. Recuperado de: <https://aws.amazon.com/es/compare/the-difference-between-deep-learning-and-neural-networks/>
- Castillo Vázquez, F. A. 2023. *Navegación autónoma de vehículos aéreos por referencia visual*. Recuperado de: https://repositorio.cetys.mx/bitstream/60000/1699/1/Tesis_Maestria_FranciscoCastillo_FINAL.pdf
- Donkeycar. s.f. *Código (imagen)*. Recuperado de: <https://docs.donkeycar.com/>
- Gómez Silva, L. D. 2023. *Implementación de un sistema de visión artificial de detección y evasión de obstáculos para un robot móvil tipo oruga*. Recuperado de: <http://hdl.handle.net/20.500.12749/23082>
- González, A. et al. (2015). *Drones aplicados a la agricultura de precisión*. *Publicaciones e Investigación*, 10(2016): 23-37. DOI: <https://doi.org/10.22490/25394088.1585>
- Pino V., E. 2019. *Los drones una herramienta para una agricultura eficiente: un futuro de alta tecnología*. *IDESIA*. 37(1): 75-84. DOI: <http://dx.doi.org/10.4067/S0718-34292019005000402>
- ROS. s.f. *ROS. Sistema operativo para robots*. Recuperado de: <https://ros.org/>
- Silva Guzmán, E. R. 2020. *Tutorial: entrenamiento de la red neuronal convolucional YOLO para objetos propios*. Recuperado de: https://biorobotics.fi-p.unam.mx/wp-content/uploads/Courses/reconocimiento_de_patrones/tutoriales/YOLO-Introducci%C3%B3n-e-implementaci%C3%B3n-.pdf
- Tovar Soto, J. P. et al. 2019. *Internet de las cosas aplicado a la agricultura: estado actual*. *Lámpsakos*, (22): 86-105. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=7210369>