RIMI
Revista de Investigación Multidisciplinaria, Iberoamericana

AÑO 2. No. 4
ISSN:2992-7978

Modelo de identificación espacial de patrones de referencia empleando Redes Neuronales Convolucionales (CNN) y entrenamiento por lotes

Mora Cabral María Teresa[1]
https://orcid.org/0009-0008-4823-9471
Camero Berrones Rosa Gabriela[2]
https://orcid.org/0000-0003-4438-1645
Arriaga Pons María Dolores[3]
https://orcid.org/0009-0007-1788-8148

[1] Doctoranda en Informática, Universidad Americana de Europa
[2] Doctora en Tecnología Avanzada, Tecnológico Nacional de México, campus Cd. Madero
[3] Doctora en Arquitectura, Universidad Juárez del Estado de Durango

Resumen

Este artículo presenta el diseño de un modelo de red neuronal convolucional (CNN) para la identificación espacial de un patrón de referencia utilizando imágenes sintéticas. En visión artificial una de las problemáticas para la detección precisa de objetos está relacionada con las condiciones de iluminación o el ruido de las imágenes, por lo que se ha propuesto un modelo de CNN que utilizó imágenes sintéticas y entrenamiento por lotes para realizar de forma precisa la identificación y ubicación espacial de un patrón de referencia. El modelo en mención se diseñó con diferentes capas de convolución y agrupamiento contando con una salida de clasificación y una de regresión. Durante las pruebas realizadas se obtuvieron valores de exactitud de arriba del 90% en la detección del patrón de referencia y una tasa de error cuadrático media relativamente baja, lo cual demostró la precisión y eficacia del modelo. Los resultados señalaron que este modelo es útil para entornos controlados, sin embargo, pudiera ser escalable y adaptable para otros entornos y condiciones variadas.

Palabras clave

Entrenamiento por lotes, Identificación de patrones de referencia, Redes Neuronales Convolucionales.

Abstract

This article presents the design of a Convolutional Neural Network (CNN) model for the spatial identification of a reference pattern using synthetic images. In computer vision, one of the challenges for accurate object detection is related to lighting conditions or image noise; thus, a CNN model was proposed, utilizing synthetic images and batch training to accurately perform the identification and spatial localization of a reference pattern. The mentioned model was designed with various convolution and pooling layers, featuring a classification output and a regression output. During testing, accuracy rates above 90% were achieved in detecting the reference pattern, along with a relatively low mean squared error rate, demonstrating the model's precision and effectiveness. The results indicated that this model is useful for

AÑO 2. No. 4
ISSN:2992-7978

controlled environments; however, it could potentially be scalable and adaptable to other environments and varying conditions.

Keyboards

Batch training, Convolutional Neural Networks, Identification of reference patterns.

INTRODUCCIÓN

En la actualidad la inteligencia artificial es una de las tecnologías con mayor impacto en diversos campos y ámbitos disciplinares, sus múltiples aplicaciones han revolucionado la forma en la cual realizamos tareas tanto profesionales como cotidianas. Una de las aplicaciones de esta tecnología es la visión artificial, cuyo objeto es lograr la capacidad de emular la visión humana, es decir, ser capaz de interpretar y comprender imágenes o grupos de imágenes del mundo real. La visión artificial es ampliamente utilizada en aplicaciones de robótica móvil, navegación autónoma, detección de rostros, diagnósticos médicos, inspección en líneas de producción entre muchas otras. Esta investigación surge de la necesidad de ubicar espacialmente un robot móvil en el mundo real, puesto que los modelos tradicionales de procesamiento de imágenes presentan problemáticas con la calibración de la cámara, así como de otros factores relacionados con la iluminación, oclusión o bien, ruido y variabilidad de las imágenes empleadas, lo cual afecta en la precisión respecto a la localización espacial del objeto en mención. Por lo anterior, se trabaja en un modelo de red neuronal convolucional (CNN) que apoyado con un patrón de referencia sea capaz de identificar y ubicar la posición de un robot móvil. El presente artículo tiene como objetivo desarrollar un modelo para la identificación espacial de un patrón de referencia circular a través de una CNN, empleando imágenes sintéticas y entrenamiento por lotes, esto último a partir de la necesidad de reducir el alto consumo de recursos computacionales que son requeridos al trabajar con modelos para procesar grandes cantidades de información, como es el caso de este proyecto. Por consiguiente, es necesario: diseñar y entrenar un modelo de CNN utilizando imágenes sintéticas para la detección de patrones de referencia, evaluar el rendimiento del modelo en la identificación de patrones bajo diferentes condiciones de ruido, iluminación y oclusión y optimizar la precisión en la localización espacial del patrón de referencia, minimizando los errores de detección. Con base en el anterior planteamiento la hipótesis de esta investigación es que con el uso de un

modelo de red neuronal convolucional (CNN), entrenado con imágenes sintéticas, es posible identificar y localizar de manera precisa un patrón de referencia circular, mejorando la precisión y el tiempo de respuesta.

ESTADO DEL ARTE

Durante los últimos años las Redes Neuronales Convolucionales (CNN) han sido empleadas principalmente para tareas de clasificación y localización de objetos en entornos complejos, dando pie al desarrollo de investigaciones para la detección y ubicación espacial a través de diferentes técnicas de entrenamiento, así como el uso de parámetros que permitan identificaciones más precisas, contribuyendo con ello en diversas áreas y disciplinas.

Cabrera Mora, 2021 desarrolló un proyecto empleando CNN para identificar y localizar un robot móvil, utilizando una salida de regresión para determinar sus coordenadas. Utilizó una CNN pre entrenada denominada Alexnet en conjunto con técnicas como data argumentation y optimización bayesiana lo cual contribuyó a lograr una mayor precisión en las predicciones del modelo. Con las pruebas realizadas concluyeron que la técnica de data argumentation fue de suma utilidad para mejorar la precisión en condiciones de iluminación limitada; por otro lado la optimización bayesiana tuvo buenos resultados, sin embargo, el tiempo de ejecución fue muy alto y no justificable ya que las mejoras no fueron significativas. Respecto a la localización del robot usando la regresión para identificar las coordenadas de ubicación del robot, señalan que no obtuvieron resultados satisfactorios ya que éstos presentan diferencias de hasta un metro respecto a la posición real del robot; por tanto, emplearon la localización jerárquica a través del uso de descriptores de apariencia global lo cual permitió conseguir errores que rondaban los 24 centímetros en condiciones de iluminación nublado y noche y de entre 61 y 69 centímetros en condiciones de iluminación soleada. Como se puede apreciar, este estudio resalta la necesidad de diseñar modelos que sean capaces de identificar y localizar patrones de referencia en entornos con condiciones de iluminación variables, siendo esto una de las problemáticas más comunes en términos de visión artificial.

En 2022, Montiel González et al. realizaron el estudio denominado: Clasificación de uso del suelo y vegetación con redes neuronales convolucionales, en el cual diseñaron un modelo para identificar patrones en la clasificación del uso de suelo y vegetación en la cuenca del rio Atoyac-Salado en Oaxaca, empleando imágenes satelitales. El modelo propuesto se entrenó utilizando datos digitales capturados en 2021 por el satélite Sentinel-2; se aplicó una combinación diferente de hiperparámetros y técnicas de regularización como Dropout, esto con el objeto de mejorar la precisión y reducir el problema de sobreajuste. Los resultados

proporcionaron una precisión de 84.57 % para el conjunto de datos, lo cual deja en evidencia que el uso de técnicas combinadas para reducir el sobreajuste, así como el ajuste de los hiperparámetros constituyen una base importante para llevar a cabo entrenamientos en entornos diversos y complejos.

En 2023, Fernández Marcos y Villarubia González de la Universidad de Salamanca, realizaron un estudio empleando la arquitectura *YOLO (You Only Look Once)* para la detección de objetos en imágenes satelitales. Llevaron a cabo un proceso de entrenamiento de varios modelos *YOLO* utilizando un conjunto de datos de imágenes satelitales clasificadas con objetos como piscinas, paneles solares y pistas de tenis, cuyo objetivo fue reconocer patrones visuales característicos de los objetos de interés y luego aplicar este modelo a nuevas imágenes para detectar y delimitar la ubicación de dichos objetos. En este estudio se resalta la importancia de la preparación de las imágenes en la etapa de preprocesamiento, (que generalmente incluye la normalización, redimensionado y etiquetado de los objetos) para que sea exitosa y precisa la identificación del objeto.

Finalmente, Gómez Pujante (2023) realizó un estudio en el área de medicina, en el cual se diseñó un modelo de CNN para la clasificación de tumores cerebrales a partir de imágenes radiográficas para el diagnóstico de neumonías. El autor utilizó diversas técnicas como *Batch Normatlization* y *Regularization Decay* para reducir la complejidad del modelo, así como Transfer Learning usando los modelos pre-entrenados: *VGG16* y *EfficientNet*, esto con el fin de mejorar la precisión, siendo evaluado a través de la métrica *accuracy* y la función de perdida *loss*. El autor destaca que la clasificación precisa de tumores cerebrales es una tarea compleja que requiere una cantidad considerable de datos de alta calidad y diversidad para capturar todas las variaciones y patrones relevantes. Para el entrenamiento del modelo, usaron *Google Colab* y encontrando algunas dificultades con el mismo, debido al tiempo que las ejecuciones eran tardadas y las limitaciones en el uso de *GPU*, lo que resultó en un tiempo de desarrollo mayor al estimado y la imposibilidad de hacer más pruebas de las deseadas.

Como puede apreciarse a través de los estudios analizados, las CNN constituyen una tecnología ampliamente utilizada para la identificación de patrones complejos y entornos diversos, además señalan la importancia de emplear diferentes técnicas que permitan optimizar los modelos y mejorar con ello la precisión y adaptabilidad de los mismos.

MARCO TEÓRICO

AÑO 2. No. 4
ISSN:2992-7978

La *inteligencia artificial (IA)* es la capacidad de una máquina de realizar tareas automatizadas que comúnmente son realizadas por un ser humano, recibiendo un conjunto de datos para su procesamiento y posteriormente utilizarlos para tomar decisiones tal como una persona lo haría. La IA cuenta con un campo de aplicación muy amplio y hoy en día se realizan múltiples investigaciones para la resolución de problemáticas cotidianas a través del uso de esta tecnología.

Uno de los campos de aplicación de la IA es la visión artificial, la cual constituye un conjunto de técnicas que permiten capturar, procesar y analizar imágenes. Su objetivo es emular la visión del ojo humano y, aunque aún dista mucho de ello, esta disciplina busca que a partir de la captura de imágenes sea posible identificar objetos y emplear dicha información en aplicaciones que permitan realizar tareas específicas como: identificación de obstáculos en la conducción autónoma, líneas de producción robotizadas, diagnóstico de enfermedades, seguridad y vigilancia, entre muchos otros.

Otra de las áreas de interés de la IA y una de las más importantes en la actualidad, es el *Machine Learning* o Aprendizaje Automático, el cual se centra en la capacidad de una computadora o una máquina de aprender por sí misma, es decir, sin la intervención explícita de un programador. Dentro del *Machine Learning*, se ubican las redes neuronales artificiales (RNA) las cuales son modelos que imitan el funcionamiento del cerebro humano, y se componen por neuronas artificiales que se encuentran conectadas entre sí en forma de capas. El objetivo de las RNA es que funcionen del mismo modo que las neuronas biológicas durante los procesos de aprendizaje, abstracción y generalización y que sean capaces de imitar a la perfección el comportamiento lógico-racional de nuestro cerebro. (Leal *et al.*, 2021). Existen diversos tipos de redes neuronales artificiales, las cuales cuentan con una gran cantidad de capas profundas, este tipo de redes y en conjunto con las distintas técnicas que se emplean para entrenarlas se denominan *Deep Learning* o Aprendizaje profundo.

El *Deep Learning* es una subárea del *Machine Learning*, tal como se muestra en la Figura 1, que a su vez constituye un subcampo de la Inteligencia Artificial, en la cual la información se procesa a través de un conjunto de capas que cuentan con una jerarquía, lo cual permite analizar y comprender dicha información de una forma profunda y creciente.

Figura 1

Subáreas de la Inteligencia artificial

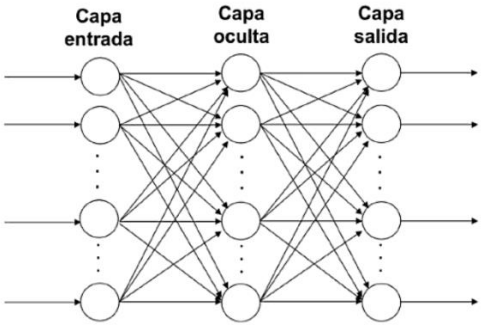
Inteligencia artificial
Diseñar y crear programas que buscan la imitación de habilidades humanas
Machine Learning
Aprender de los datos sin ser programadas explícitamente
Deep Learning:
Redes neuronales artificiales, inspiradas en el funcionamiento del cerebro humano

Nota: Elaboración propia

En general todos los algoritmos de *Deep Learning* son redes neuronales, puesto que están constituidos por un conjunto de capas de neuronas que se encuentran interconectadas. Una red neuronal se considera *Deep Learning* cuando tiene una o más capas ocultas. Consulte la Figura 2 la cual muestra una representación gráfica de una red neuronal profunda.

Figura 2

Representación de una red neuronal profunda

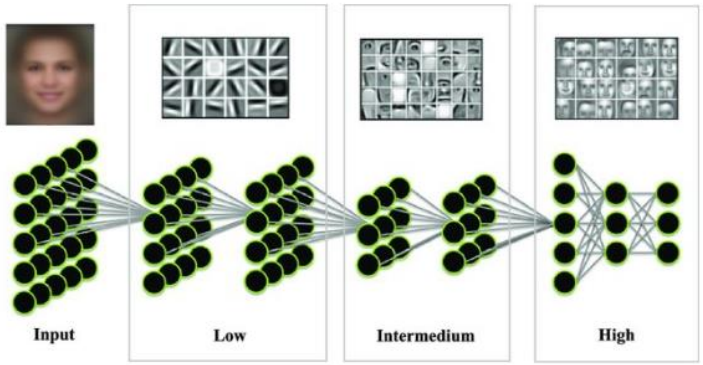


Nota: Tomado de Torres (2018)

Una red neuronal convolucional es un tipo de algoritmo de aprendizaje profundo que se utiliza mayormente para analizar y aprender atributos visuales de grandes cantidades de datos. (Intel Corporation, 2023). Son un tipo de red neuronal especializada en el procesamiento de datos y están compuestas por varios tipos de capas especiales; su característica principal es que éstas son orientadas principalmente a recibir entradas de tipo imágenes, lo cual permite que puedan ser reconocidos elementos u objetos de ellas. Este tipo de redes neuronales son mayormente utilizadas para el área de visión artificial. La Figura 3, muestra un ejemplo de la representación gráfica de una CNN.

Figura 3

Ejemplo de la representación gráfica de una red neuronal convolucional



Nota: Tomado de Li *et al.* (2019)

Para poder llevar a cabo la identificación de objetos una red neuronal convolucional puede estar integrada de diferentes tipos de capas, las cuales tienen funcionalidades específicas que le permiten determinar las características necesarias para poder llevar a cabo la toma de decisiones y determinar la salida correspondiente a cada modelo. Las principales capas que comúnmente componen una CNN son las siguientes:

Capa de convolución *(Convolution layer)*

Las capas convolucionales se entienden como un conjunto de filtros comúnmente llamados: campos receptivos, éstos se ajustan para la extracción de características de una señal. (Cifuentes et al., 2019), el propósito de la convolución es la extracción de características de la imagen de entrada.

Capa de agrupamiento (*subsampling, pooling*)

Esta capa recibe la información generada en la capa de convolución, reduce la imagen a la mitad y selecciona solo aquellas características más importantes, descartando el resto. En esta capa se decrementa el tamaño de las imágenes con el objetivo de mejorar la eficiencia computacional. (Raschka & Mirjalili, 2019)

Capa totalmente conectada (*Fully connected layer*)

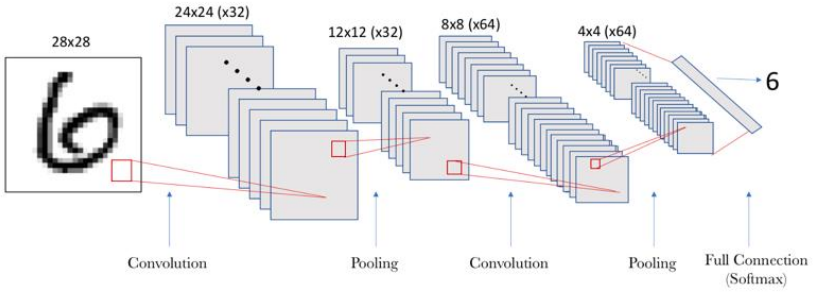
El resultado de las capas convolucionales y de agrupación representan características de alto nivel de la imagen de entrada. El propósito de la capa totalmente conectada es utilizar estas características para catalogar la imagen de entrada en varias clases según el conjunto de datos de entrenamiento. (Sánchez-Alor Expósito, 2020)

En la Figura 4, se representa una CNN, puede apreciarse que se encuentra compuesta por distintas capas alternadas de convolución y agrupamiento, precedidas por una capa completamente conectada, la

cual como ya se ha mencionado se encarga de la clasificación de la imagen de acuerdo con las clases que puede reconocer y que fueron previamente entrenadas.

Figura 4

Diagrama de una CNN



Nota: Tomado de Torres (2018)

Con el objeto de probar el funcionamiento correcto de una red neuronal, es requerido evaluar su rendimiento haciendo uso de métricas de evaluación, las cuales brindan una estimación contemplando diversos criterios que permitan determinar el buen funcionamiento de la red neuronal. Para llevar a cabo este proceso es necesario que se tengan los resultados del modelo que emanan de los procesos de entrenamiento, validación y prueba.

Para este modelo, una de las métricas usadas es la Exactitud o *Accuracy* la cual muestra el ratio de datos clasificados correctamente (Borrero y Arias, 2021). Es una de las métricas más comunes para los problemas de clasificación y ha sido empleada principalmente para evaluar el rendimiento de la salida binaria de este modelo. Está definida de la siguiente manera:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

De acuerdo con Borrego y Arias (2021), el Error Cuadrático Medio (EMC) o *mean square error*, generalmente es empleado en problemas de regresión. Consiste en la media de las diferencias al cuadrado entre la salida esperada y la salida obtenida. Esta función se define de la siguiente manera:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - z_i)^2$$

Donde "n" representa el número de elementos que intervienen en el pronóstico, "y_i" representa el valor real y "z_j" representa la salida pronosticada. (Carpio y Valdivieso, 2020)

AÑO 2. No. 4
ISSN:2992-7978

METODOLOGÍA

Descripción del modelo

Este proyecto fue desarrollado para detectar un patrón de referencia circular empleando redes neuronales convolucionales, el modelo fue diseñado para llevar a cabo dos tareas: la primera consiste en realizar una clasificación binaria la cual determina si la imagen que se evaluó tiene un círculo o no, la segunda consistió en predecir las coordenadas de la ubicación del círculo. El modelo de CNN se integró por distintas capas cuya función se describe a continuación:

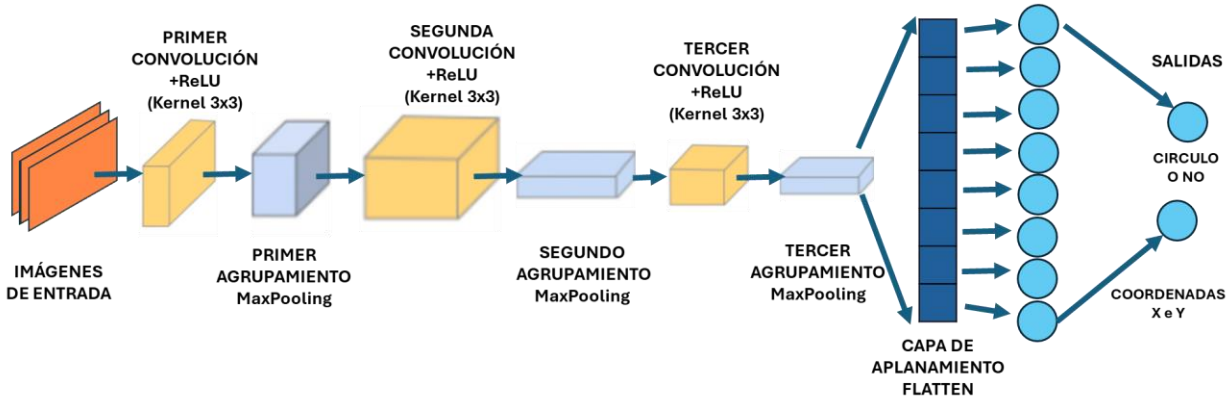
- **Capa de entrada:** La primera capa del modelo recibió las imágenes sintéticas a analizar en escala de grises con una dimensión de 240x180 pixeles.
- **Capas de convolución:** El modelo se diseñó con tres capas de convolución las cuales se encargaron de extraer las características principales de las imágenes. La primera capa empleó 32 filtros con un tamaño de 3x3 para extraer las características básicas, en la segunda capa se utilizaron 64 filtros de tamaño 3x3 lo que permitió ir detectando características más complejas de las imágenes y en la tercera se emplearon 128 filtros de igual tamaño que las capas anteriores, aumentando el grado de complejidad y abstracción de las características extraídas. En cada una de estas convoluciones se utilizó la función de activación *ReLU*.
- **Capas de agrupamiento:** Por cada capa de convolución se agregó una capa de agrupamiento, las cuales se encargaron de reducir las dimensiones de la imagen que se obtuvo de entrada a una región de 2x2 pixeles, esto permitió disminuir la complejidad del modelo, manteniendo únicamente las características más importantes detectadas en las capas de convolución, de esta manera se facilitó el procesamiento buscando no caer en la sobrecarga del modelo gracias a la reducción de parámetros.
- **Capa de aplanamiento:** En esta capa se transformaron los datos bidimensionales extraídos en las capas de convolución a un vector unidimensional, el objeto de este proceso es que dichos datos puedan conectarse con la capa densa que únicamente trabaja con vectores. La capa *Flatten* o de

aplanamiento tuvo como objetivo primordial convertir los datos a una estructura que sea compatible con las capas completamente conectadas.

- **Capas completamente conectadas:**
  - **Capa densa:** Dentro de las capas completamente conectadas, se agregaron una capa densa de 64 unidades con una activación *ReLu*, esto permitió a la red neuronal combinar las características que fueron extraídas de las imágenes para aprender patrones más complejos. Esta es una de las capas responsables de usar toda la información extraída de las capas anteriores para realizar la clasificación o la predicción para lo cual fue creado el modelo, es decir, constituyen la parte final de la red en la cual se interpretan las características que fueron aprendidas durante el proceso de entrenamiento.
  - **Capa Dropout:** No es como tal una capa, pero se aplica dentro de las *Fully Connected Layers*, para evitar la interdependencia de las neuronas y forzar a la red a aprender representaciones más robustas. Para este caso, fue configurada con una probabilidad de 0.5, lo cual permitió que durante la etapa de entrenamiento el 50% de las neuronas fueran apagadas de forma aleatoria en cada iteración con el objetivo de evitar el sobreajuste y de esta manera mejorar la generalización del modelo.
- **Capa de salida:** El modelo se diseñó para regresar dos salidas, una determinó si las imágenes evaluadas contenían o no un círculo, la segunda predijo las coordenadas en pixeles "x" e "y" de la posición del círculo en la imagen analizada. En la Figura 5 se muestra gráficamente el modelo descrito.

Figura 5

Modelo de CNN para la detección espacial de un patrón de referencia.



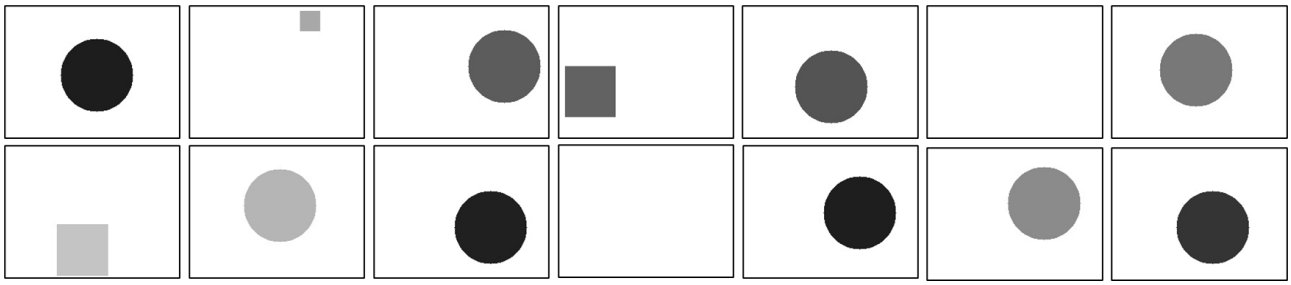
Nota: Elaboración propia.

Preparación de datos

Una vez desarrollado el modelo se procedió a preparar los datos para llevar a cabo el entrenamiento, para esto se generó un Dataset con ayuda de una función que creó imágenes sintéticas con dimensiones de 240x180 pixeles en escala de grises. Se realizaron pruebas con diferentes tamaños de *dataset*, siendo el más pequeño de 400 y el más grande de 2000 imágenes sintéticas, de las cuales se utilizaron 70% para el entrenamiento y 30% para pruebas en cada caso. Estos datos de entrada incluyeron círculos o cuadrados en diferentes posiciones dentro de la imagen o bien, imágenes en blanco, con el propósito de brindar al modelo ejemplos diversos que le permitieran encontrar las características particulares de la figura de referencia que para este caso fue un círculo. En la Figura 6 se presenta algunos ejemplos de las imágenes que se introdujeron durante el proceso de entrenamiento:

Figura 6

Ejemplos de imágenes del dataset para el entrenamiento del modelo CNN



Nota: Elaboración propia.

Entrenamiento y pruebas

Trabajar con *Deep Learning*, específicamente para este caso con redes neuronales convolucionales, representó un alto consumo de recursos computacionales, sobre todo y debido a la cantidad y tamaño de las imágenes que componen el *dataset* para el proceso de entrenamiento y prueba. Si el *dataset* está compuesto por un gran número de imágenes, se incrementará indudablemente el costo de procesamiento y la memoria *RAM*, por consiguiente, equipos con limitaciones computacionales de procesamiento y memoria *RAM*, difícilmente podrán ejecutar el modelo de forma óptima o incluso no lograrán la ejecución del mismo.

AÑO 2. No. 4
ISSN:2992-7978

Para esta etapa de la evaluación del modelo se realizaron pruebas con diferentes tamaños de *datasets* variando de entre 200 a 2000 imágenes sintéticas y realizando un total de 50 épocas. El equipo en el cual se realizaron las pruebas cuenta con las siguientes características de hardware: AMD Ryzen 5 2500U, con Radeon Vega Mobile Gfx, 2000 Mhz, 4 procesadores principales y 8 procesadores lógicos, así como 12 GB de memoria RAM; sin embargo, la ejecución del modelo tuvo inconvenientes respecto al alto consumo de los recursos, siendo complejo lograr la ejecución óptima del mismo.

De acuerdo con ASUS (2024), los requisitos mínimos de hardware computacional para el trabajo con Redes Neuronales Convolucionales son:

- Procesador de 16 a 24 núcleos a 5GHz o más cuando está potenciado.
- GPU con una memoria de video de mínimo 8GB
- Memoria RAM se recomienda contar con el doble de memoria VRAM de la GPU, por consiguiente, es requerido como mínimo 16GB de RAM, preferentemente tecnología DDR5
- 1TB como mínimo de almacenamiento SSD

Como puede apreciarse, el equipo computacional con el cual se llevaron a cabo las pruebas era limitado comparado con la información antes presentada, por tal motivo, durante el desarrollo de las pruebas se enfrentó a la dificultad de la ejecución del modelo a causa de los bajos recursos de hardware computacional con el cual se contaba, por lo cual se diseñó una solución de software con el propósito de que el proceso de entrenamiento pudiera ser realizado en el equipo de cómputo de pruebas, entrenando el modelo en lotes o *batches.*

El entrenamiento por lotes consistió en dividir el *dataset* en subconjuntos más pequeños de imágenes que fueron alojadas en una base de datos en *MySQL*, con el objeto de aligerar la carga de la memoria *RAM* y el procesador del equipo en mención. Se llevaron a cabo diversas pruebas con diferentes tamaños de *datasets* y *batches*, llegando a probarlo con hasta 2000 imágenes de las cuales 1400 se emplearon para el entrenamiento y 600 se usaron para el proceso de validación, así mismo se configuró como tamaño del *batch* 300 imágenes y se realizaron 50 épocas. Por último, cabe mencionar que para probar el modelo se utilizaron 10 imágenes sintéticas que de forma aleatoria fueron generadas con o sin el patrón de referencia que el modelo debía identificar. Con los parámetros antes descritos la ejecución del modelo incluyendo el entrenamiento, la validación y las pruebas se llevó a cabo aproximadamente en una hora y media, logrando concluir de manera satisfactoria.

RESULTADOS

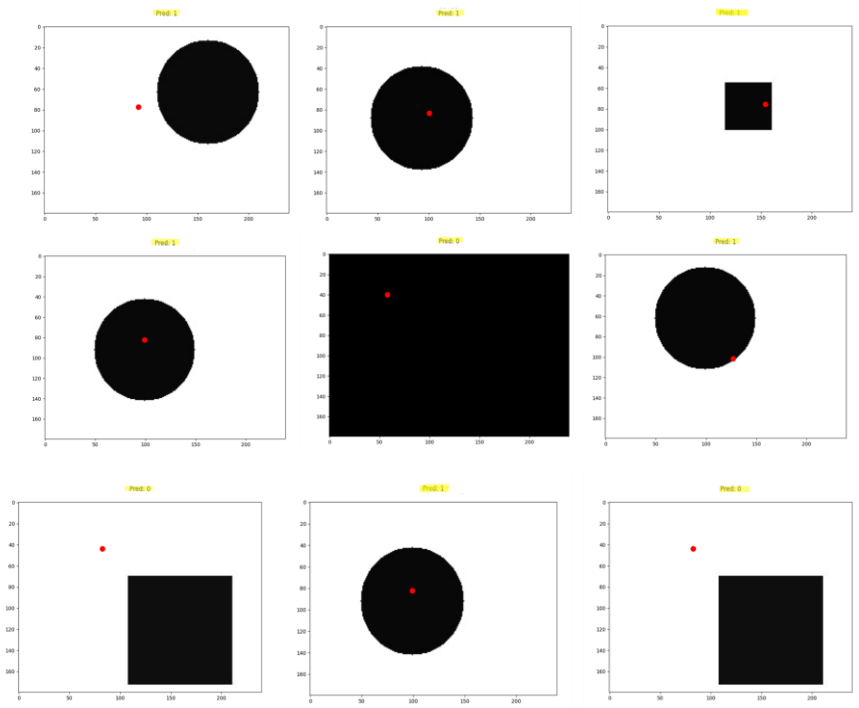
Después del entrenamiento y prueba del modelo se obtuvieron dos salidas: Una clasificación binaria que determina si la imagen contiene o no el patrón de referencia y la salida de predicción de coordenadas cuyo propósito es identificar las coordenadas x e y en pixeles de la figura de referencia localizada en la imagen.

Para mostrar los resultados de la clasificación binaria se emplea la variable *Pred* que muestra un 1 si la imagen tiene el patrón de referencia y un 0 si determina que dicho patrón no existe en la imagen. Respecto a la salida de la predicción de las coordenadas en cada imagen se resalta dicho resultado con un punto en color rojo.

En la Figura 7, se muestran los resultados de las imágenes que se usaron para las pruebas. Como puede apreciarse el modelo predice y muestra si la imagen contiene o no el patrón de referencia y muestra con un punto en color rojo la predicción de las coordenadas x e y de dicho patrón.

Figura 7

Resultados de las pruebas realizadas



Nota: Elaboración propia.

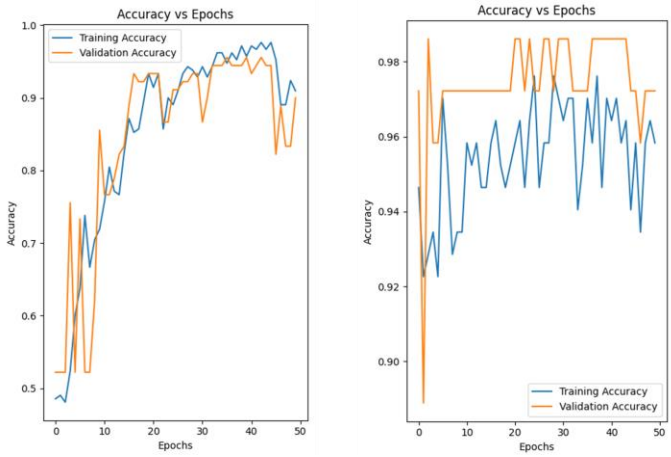
AÑO 2. No. 4
ISSN:2992-7978

Para medir la eficacia del modelo se emplearon principalmente la métrica Accuracy o Exactitud y la función de pérdida Mean Square Error (MSE).

Para conocer la precisión de la clasificación binaria, es decir, si el modelo es preciso al determinar si el patrón de referencia existe o no en una imagen, se empleó la métrica *Accuracy*. En la Figura 8, se muestran los resultados graficados de esta métrica, si bien fueron generadas varias gráficas durante el proceso en función del número de *batches* creados, en este apartado únicamente se incluyen los resultados del primer y el último *batch*.

Figura 8

Resultados de la métrica Accuracy



Nota: Elaboración propia. Gráfica izquierda generada al finalizar el primer batch de entrenamiento. Gráfica derecha generada al concluir el último batch de entrenamiento.

Consulte en la Figura 8 resaltado en color azul el resultado obtenido durante el entrenamiento, y en color naranja la información generada del proceso de validación. En la gráfica de la izquierda, se observa que al evaluar la precisión para el caso del proceso de entrenamiento existen fluctuaciones en las primeras épocas, sin embargo, se aprecia una mejora progresiva siendo esto un indicador importante que refiere que el modelo está aprendiendo a ajustarse a los datos de entrenamiento. Por su parte, respecto a la validación de igual manera se visualizan altas fluctuaciones en las primeras épocas, sin embargo, esto puede ser a causa de los intentos que hace el modelo para realizar generalizaciones con poca información, es decir, aún no aprende las características más importantes del objeto a identificar, no obstante, este valor presenta mejoras a lo largo de las épocas, lo cual indica que el modelo poco a poco aprende a generalizar.

Al final del último *batch* entrenamiento, en la gráfica de la derecha se puede visualizar una marcada estabilidad dado que las fluctuaciones disminuyen considerablemente tanto en el proceso de entrenamiento

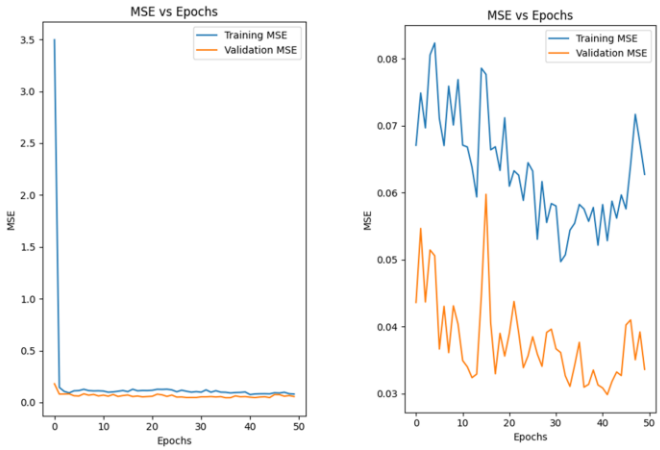
como en el de validación, observe que los valores se sitúan entre 0.90 y 0.98, lo cual es muy notable respecto al inicio del entrenamiento cuyos valores se posicionaban entre 0.5 y 0.9. Los valores obtenidos en esta fase final indican que el modelo ha aprendido a generalizar de forma adecuada los datos, si bien hay pequeñas fluctuaciones, los valores son muy cercanos al 100% de precisión. La estabilidad final que se observa en esta métrica señala que el modelo realiza la clasificación binaria de forma correcta y confiable.

Para el caso de la salida de predicción de las coordenadas, se emplea la función Error Cuadrático Medio o MSE, ya que generalmente es útil para el desarrollo de modelos de redes neuronales convolucionales en los cuales es necesario determinar de forma precisa valores continuos, como coordenadas, (siendo el caso de este modelo).

De igual manera, se generan dos gráficas, una que mide el grado de error al finalizar el primer *batch* de entrenamiento y otra que lo hace al final. Esto se ilustra en la Figura 9.

Figura 9

Resultados de la función MSE



Nota: Elaboración propia. Gráfica izquierda generada al finalizar el primer batch de entrenamiento. Gráfica derecha generada al concluir el último batch de entrenamiento

En la gráfica de la izquierda, se aprecia el resultado de la función de error, observe que, en las primeras épocas del entrenamiento, el valor del MSE es muy elevando, aproximadamente 3.5, lo anterior es un indicativo de que el modelo tiene un bajo rendimiento en ese momento, lo cual es normal dado que el modelo aún está en una fase inicial de aprendizaje. No obstante, y de forma muy abrupta el valor de MSE disminuye en las primeras cinco épocas; esta rápida disminución solo indica que el modelo ajusta sus parámetros de

forma rápida con el objetivo de mejorar el rendimiento. En el proceso de validación se puede observar un comportamiento similar.

Por su parte, la gráfica de la derecha que muestra el resultado de MSE al final del proceso de entrenamiento, observe que, si bien el valor sigue disminuyendo lo hace a un ritmo más lento a diferencia de las épocas iniciales, ya que dicho valor varía entre 0.8 y 0.5. Observe que el MSE del proceso de entrenamiento, (el cual se muestra en color azul) muestra un patrón de disminución progresiva, aunque con algunas ligeras fluctuaciones. Por otra parte, el MSE del proceso de validación (representado en color naranja en la gráfica), de igual manera disminuye, aunque se aprecia un valor ligeramente más bajo y más estable que oscila entre 0.6 y 0.3, respecto al proceso de entrenamiento; esto es un indicativo de que el modelo generaliza de forma correcta.

DISCUSIÓN

El modelo presentado en este documento representa un diseño muy específico para la detección de patrones de referencia que pueda ser empleado para la ubicación espacial de un objeto, siendo adecuado para entornos controlados o áreas delimitadas.

De acuerdo con el modelo diseñado y con base en los resultados obtenidos se reconoce como efectivo para detectar el patrón de referencia para el cual fue entrenado, ya que es capaz de obtener resultados satisfactorios en el 90% de los casos. Respecto a las predicciones sobre las coordenadas x e y del objeto identificado, si bien las métricas MSE son alentadoras, la precisión aún no es muy certera por lo que es necesario establecer mejoras en el modelo, podrían explorarse optimizaciones adicionales como la normalización en las capas convolucionales y el ajuste de hiperparámetros para mejorar la precisión y reducir el error en la predicción de las coordenadas. Así mismo, cabe hacer mención que en otros estudios como el de Cabrera (2021), ha ocurrido una situación similar, puesto que este autor señala no haber obtenido resultados satisfactorios prediciendo las coordenadas por un método de regresión, por lo cual es necesario explorar alternativas como la localización jerárquica, señalada en dicho estudio.

Una de las problemáticas a las cuales es posible enfrentarse al trabajar con CNN, es el alto consumo computacional que en algunas ocasiones puede ser limitante debido a las características del equipo de cómputo con el cual se realicen las pruebas. Ante este problema, que se ha señalado en el estudio de Gómez (2023), en el cual indica haber empleado *Google Colab* y encontrar aun así dificultades debido al tiempo de ejecución o limitantes en el uso de GPU, esta propuesta emplea el entrenamiento por lotes, el cual permite

dividir los conjuntos de datos de entrada en bloques más pequeños con el fin de que equipos computacionales con limitantes de hardware sean capaces de ejecutar el modelo completamente. Es necesario seguir probando este modelo en equipos computacionales que de igual manera cuenten con recursos limitados, con el objeto de medir su eficiencia y rendimiento de acuerdo con las distintas características computacionales que puedan tener.

CONCLUSIONES

El modelo de CNN desarrollado cuenta con la capacidad de identificar espacialmente un patrón de referencia de una forma óptima empleando imágenes sintéticas y entrenamiento por lotes; lo anterior establece las bases para llevarlo a un entorno de pruebas que permita identificar espacialmente un robot móvil utilizando el patrón de referencia de este modelo. Si bien el modelo ha presentado reducción del error y aumento de la precisión en la clasificación, también presenta retos que deben abordarse para garantizar un modelo robusto y generalizable.

En futuros trabajos, se pretende lograr la expansión del modelo para identificar otros tipos de patrones o figuras y mejorar la precisión en entornos más complejos, así como emplear el modelo utilizando estos patrones de referencia para la ubicación de otros objetos como robots móviles, lo anterior puede establecer mejoras para la generación de rutas de navegación por ejemplo en una fábrica o almacén, podría emplearse en el ámbito de agricultura de precisión identificando patrones en los campos que permitan llevar a cabo por ejemplo tareas de fumigación o monitoreo de cultivos. En general el modelo puede ser escalable y adaptable en diferentes sectores, ya que es flexible y puede adecuarse para detectar otro tipo de patrones de referencia, contribuyendo al campo de visión artificial.

REFERENCIAS

ASUS. (2024, junio). ¿Cómo elegir una laptop para trabajar con IA y Aprendizaje Automático? ASUS México. https://www.asus.com/mx/content/how-to-choose-a-computer-for-ai-and-machine-learning-work/

Borrero, I. P., y Arias, M. E. G. (2021). Deep learning. Servicio de Publicaciones de la Universidad de Huelva.

Cabrera Mora, J. J. (2021). Entrenamiento, optimización y validación de una CNN para la localización de un robot móvil mediante tareas de clasificación y regresión [Universidad Miguel Hernández de Elche]. https://dspace.umh.es/bitstream/11000/26485/1/TFG-Cabrera%20Mora%2c%20Juan%20Jos%c3%a9.pdf

Carpio, K. P., y Valdivieso, F. O.-. (2020). Redes neuronales artificiales aplicadas en sistemas de predicción para la seguridad vial. Avances Investigación en Ingeniería, 17(2 (Julio-Diciembre)), Article 2 (Julio-Diciembre). https://doi.org/10.18041/1794-4953/avances.2.6632

Cifuentes, A., Mendoza, E., Lizcano, M., Santrich, A., & Moreno-Trillos, S. (2019). Desarrollo de una red neuronal convolucional para reconocer patrones en imágenes. Investigación y Desarrollo en TIC, 10(2), 7–17.

Fernández Marcos, S., & Villarubia González, G. (2023). *Sistema de reconocimiento de patrones en imágenes satelitales para la detección de objetos* [Universidad de Salamanca]. https://gredos.usal.es/bitstream/handle/10366/158423/Memoria.pdf?sequence=1&isAllowed=y

Gómez Pujante, B. (2023). Redes Convolucionales. Aplicación a la clasificación de imágenes médicas. https://dspace.umh.es/bitstream/11000/30233/1/TFG-G%C3%B3mez%20Pujante%2C%20Bego%C3%B1a.pdf

Intel Corporation. (2023). Redes neuronales convolucionales (CNN) y aprendizaje profundo [Corporativa]. Intel. https://www.intel.com/content/www/es/es/internet-of-things/computer-vision/convolutional-neural-networks.html

Leal, D. A., Restrepo, Porto, J. P., Viloria, & Algarín, C. A., Robles. (2021). El camino a las redes neuronales artificiales. Editorial Unimagdalena.

Li, X., Liu, Z., Cui, S., Luo, C., Li, C.-F., & Zhuang, Z. (2019). Predicting the effective mechanical property of heterogeneous materials by image based modeling and deep learning. Computer Methods in Applied Mechanics and Engineering, 347. https://doi.org/10.1016/j.cma.2019.01.005

Montiel González, R., Bolaños González, M. A., Macedo Cruz, A., Rodríguez González, A., & López ´Pérez, A. (2022). Clasificación de uso del suelo y vegetación con redes neuronales convolucionales. Revista Mexicana de Ciencias Forestales, 13(74), Article 74. https://doi.org/10.29298/rmcf.v13i74.1269

Sánchez-Alor Expósito, J. (2020). Evaluación de algoritmos de detección de objetos basados en deep learning para detección de incidencias en carreteras. Universidad de Valladolid.

Raschka, S., y Mirjalili, V. (2019). Python Machine Learning (3rd ed.). Packt Publishing Ltd.

Torres, J. (2018). Deep Learning Introducción práctica con Keras. Lulu.com.