

**“Procesamiento de imágenes para la clasificación de 10 tipos de felinos por medio de técnicas de computer vision y redes neuronales convolucional con Python”**

**Autor**: Estudiante. Rios, Leandro Agustin.

**Institución**: Colegio Universitario IES 21.

**Carrera**: Tecnicatura superior en ciencia de datos e inteligencia artificial.

**Materia**: Procesamiento de imágenes.

**Docente**: Buteler, Ramiro Ariel.

Córdoba, Argentina.

2023

# 1.Introducción

Una empresa especializada en la fabricación de cámaras de foto trampa, dispositivo automático usado para capturar imágenes fotográficas de animales en estado salvaje, está buscando desarrollar un algoritmo que pueda detectar y clasificar diferentes especies de felinos a partir de las imágenes capturadas.

Ampliando el contexto, las reservas naturales que compran el producto a esta empresa han notado una gran disminución en la población de felinos, por lo cual han propuesto a la empresa agregarle algún método que facilite la detección de felinos en las miles de fotos que toman, agilizando de esta forma el proceso de clasificación y permitiendo que los investigadores de la reserva se puedan enfocar mejor en las imágenes que realmente les interesan. Gracias a esto ellos no tendrían que ver las imágenes una a una sino solamente la de los felinos. Al poder hacer esto, la misma cámara mandará una señal a la guardia del parque para alertar sobre esta detección de los animales y poder movilizar a los investigadores. Se busca con este método poder descifrar más fácil el motivo por lo que la población de estos animales está decayendo.

Para lograr este objetivo, se está trabajando en la creación de un programa de aprendizaje automático que pueda clasificar cada imagen de felino según su categoría. El programa utilizará técnicas de computer vision para la detección del animal y redes neuronales convolucionales para poder descartar las imágenes que le son relevantes a los investigadores.

El objetivo planteado es:

* Desarrollar un algoritmo de clasificación de felinos en imágenes utilizando una red neuronal convolucional, aplicando técnicas avanzadas de machine learning y computer vision. El algoritmo debe alcanzar un nivel de eficacia de al menos el 85% en la precisión de la clasificación. El proyecto tiene un plazo de 3 meses y debe estar finalizado antes de julio de 2023.

# 2. Marco teórico

En este capítulo se describen los fundamentos teóricos para la comprensión del trabajo. En un inicio se aborda la definición de visión por computadora y la clasificación de imágenes, también, vamos a definir qué es el deep learning y luego, un red neuronal convolucional.

Citando la definición sobre lo que es el computer vision que IBM nos da en su página web(2023)(2). “La visión por computadora es un campo de la inteligencia artificial (IA) que permite que las computadoras y los sistemas obtengan información significativa de imágenes digitales, videos y otras entradas visuales, y tomen medidas o hagan recomendaciones basadas en esa información…”.

Cuando clasificamos una imagen tenemos un diagrama que nos puede servir para ir paso por paso, vamos a explicar cada una de las fases del siguiente diagrama. (López-Saca, Fidel, 2019)

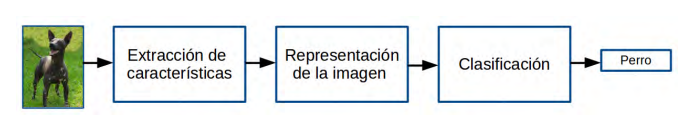


Figura 2.1: Diagrama general de identificación de imágenes.

* **Extracción de características:** Las características son los elementos que describen un objeto, como el tamaño, la forma, el color, etc. La extracción de características es un factor importante para lograr un buen rendimiento. Normalmente las características más representativas se encuentran localizadas en zonas con cambios de contraste o en los contornos de los objetos.
* **Representación de la imagen:** Desde el punto de vista estadístico, se realiza una representación única de toda la imagen en forma de arreglo numérico. Esta representación se obtiene sumando o combinando cada uno de los vectores numéricos que describen a cada uno de los puntos de interés
* **Clasificación:** se realiza la clasificación de la imagen con base en su representación única. Es en esta etapa donde el sistema de clasificación aprende del conjunto de entrenamiento a distinguir las imágenes de las diferentes categorías a partir de la representación visual de cada imagen.

En esta parte de la teoría, nos toca hablar de lo que es el deep learning, el mismo lo podemos definir de la siguiente forma “un tipo de machine learning que entrena a una computadora para que realice tareas como las hacemos los seres humanos, como el reconocimiento del habla, la identificación de imágenes o hacer predicciones. En lugar de organizar datos para que se ejecuten a través de ecuaciones predefinidas, el deep learning configura parámetros básicos acerca de los datos y entrena a la computadora para que aprenda por cuenta propia reconociendo patrones mediante el uso de muchas capas de procesamiento.” según la página Sas.com (2017).

Entonces para poder trabajar con las imágenes como nosotros necesitamos, vamos a estar usando la red neuronal convolucional (RNC) o Convolutional Neural Network (CNN). Como vamos a ver en la siguiente figura, este tipo de redes se encuentran en el área del deep learning ya que están diseñadas en forma de redes neuronales profundas.

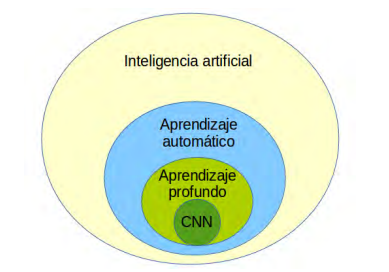


Figura 2.2: Organización estructural de la IA, lugar donde se encuentra la CNN

Citamos una parte del documento de López-Saca, Fidel en el cual dice que “Las CNN se inspiraron en la estructura del sistema visual. Al día de hoy se encuentran entre los sistemas con mejores resultados para el reconocimiento de patrones y son utilizados para el procesamiento de imágenes, vídeo, voz y audio. Estas redes han demostrado ser una herramienta posible de implementarse en distintas áreas.

La CNN es una red neuronal profunda, tiene entre sus capas a la operación de convolución que realiza una operación entre las matrices de la imagen y una matriz llamada kernel, utilizada para obtener las características como son borde, enfoque, entre otras.”

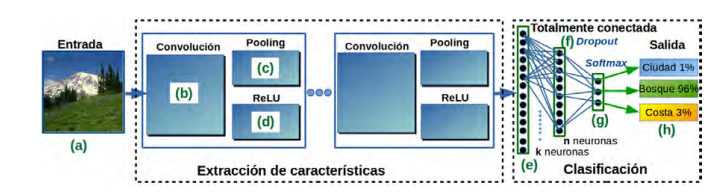


Figura 2.3: Arquitectura y proceso que sigue la CNN

Basándonos nuevamente en el documento de López-Saca, Fidel los elementos que componen las CNN son los siguientes: entrada, convolución, pooling, ReLU, una capa totalmente conectada(que pueden ser una o más) y por último la capa de salida. La capa de convolución y de reducción pueden repetirse varias veces, ahí dependen del modelo y son usadas para obtener las características más representativas de la imagen.

La normalización de imágenes es una técnica esencial que escala los valores de los píxeles de una imagen al mismo rango de valores. Al aplicar esta técnica, se logra mejorar la velocidad de entrenamiento, la precisión de clasificación y la estabilidad del proceso de entrenamiento de redes neuronales. Esta práctica consiste en dividir cada valor de píxel por la desviación estándar de la imagen y luego sumar la media de la misma. De esta manera, se garantiza que los valores de los píxeles estén en un rango uniforme, lo que favorece un aprendizaje más rápido y preciso por parte de las redes neuronales. Además, la normalización de imágenes contribuye a una mayor estabilidad durante el entrenamiento, al reducir la sensibilidad de los pesos de la red neuronal a los cambios en los datos de entrenamiento. Esto asegura que los pesos se ajusten adecuadamente, independientemente de los valores de los píxeles presentes en los datos utilizados para entrenar la red.(Adegoke, 2018)

# 3. Marco metodológico

El problema que debemos abordar es la detección de felinos en las imágenes proporcionadas por las cámaras de foto trampa. Para ello, se nos ha facilitado un conjunto de datos de imágenes con el cual realizaremos las pruebas necesarias.

El conjunto de datos se compone de la siguiente forma:

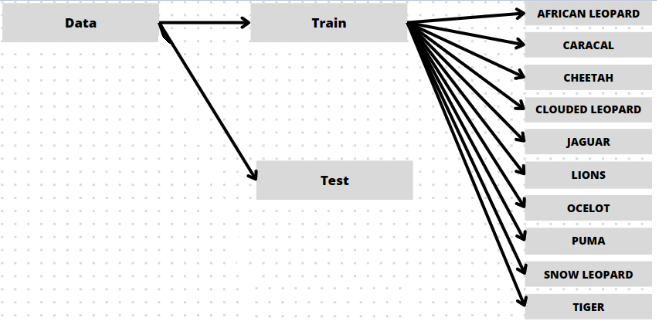


Figura 3.1: Diagrama de organización de los datos.

Existe una carpeta principal llamada "data" que contiene dos subcarpetas: "train" y "test". Cada una de estas subcarpetas contiene diez carpetas, representando cada una una categoría específica para la clasificación. Cada carpeta alberga un total de 236 imágenes, con un peso promedio de 23 KB por imagen. Esta organización estructurada de las imágenes en sus respectivas categorías contribuye a un entrenamiento más efectivo y eficiente de las redes neuronales convolucionales, lo que se traduce en un mejor rendimiento y una mayor capacidad de generalización del modelo.

Uno de los aspectos fundamentales radica en las bibliotecas con las que trabajaremos, ya que nos permiten reutilizar código preexistente, agilizando el desarrollo y asegurando una mayor eficiencia en nuestro proyecto. Las bibliotecas que utilizaremos para el manejo de datos y visualizaciones serán pandas, numpy, matplotlib, random y os. En cuanto al entrenamiento de la red neuronal, haremos uso de tensorflow y keras. Estas poderosas herramientas nos permitirán realizar tareas complejas de manera más efectiva y facilitarán la implementación de nuestro algoritmo de clasificación de felinos en imágenes.

Hablemos ahora sobre la fase de entrenamiento de los modelos, en la que trabajaremos con dos enfoques distintos. El primero implica el entrenamiento de un modelo con aumentación de datos, mientras que el segundo no emplea esta técnica. Utilizaremos la métrica de precisión (Accuracy) para evaluar el rendimiento de ambos modelos y visualizamos las curvas de aprendizaje para analizar los resultados obtenidos.

Comencemos hablando del modelo sin aumento de datos, en esta primera instancia lo que hacemos con las imágenes es normalizar los pixeles para que queden en un rango de entre 0 y 1. Por el lado del otro modelo, el que tiene aumentación de datos, usamos lo mismo para normalizar pero también agregamos rotación horizontal aleatoria, rotación vertical, les aplicamos zoom pero limitando solamente a un 5 % de la imagen, ya se para aumentar o disminuir el tamaño y por último utilizamos rotación de 25% para ir girando la imagen cada noventa grados.

| **Modelo** | **Preprocesamiento de datos** | **Métrica de evaluación** |
| --- | --- | --- |
| Sin aumento de datos | Normalización de pixeles (0 a 1) | Accuracy |
| Con aumento de datos | Normalización de pixeles (0 a 1) + Aumentación de datos (rotación horizontal, rotación vertical, zoom y rotación de 25%) | Accuracy |

Después de realizar esa etapa, procedemos a cargar y generar lotes de imágenes desde un directorio. Esto nos permitirá aplicar la normalización a todas las imágenes y utilizar estos lotes para el entrenamiento y la prueba.

En cuanto a la arquitectura de la red, utilizaremos el mismo diseño para ambos modelos. Definiremos una red neuronal convolucional mediante la API funcional de Keras en TensorFlow. A continuación, se mostrará una imagen de la arquitectura con su respectivo código, para luego proceder a una explicación detallada de cada parte.

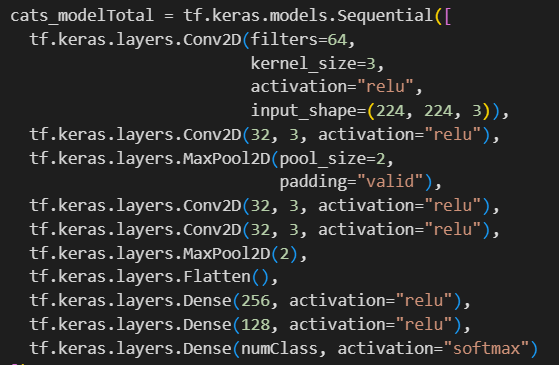


Figura 3.2: Arquitectura interna de la red neuronal de convolución.

Conv2D: cuenta con 64 filtros, un kernel de 3x3 y su función de activación es una ReLU.

1. Conv2D: cuenta con 32 filtros, un kernel de 3x3 y su función de activación es una ReLU.
2. MaxPool2D: esta capa de agrupamiento máximo realiza un muestreo espacial reduciendo a la mitad el tamaño de las dimensiones espaciales de la entrada. Utiliza un tamaño de agrupamiento de 2x2 y el modo de relleno "valid".
3. Conv2D: una tercera capa convolucional con 32 filtros, un tamaño de kernel de 3x3 y una función de activación ReLU.
4. Conv2D: Una cuarta capa convolucional con 32 filtros, un tamaño de kernel de 3x3 y una función de activación ReLU.
5. MaxPool2D: Otra capa de agrupamiento máximo con un tamaño de agrupamiento de 2x2.
6. Flatten: Esta capa aplana la salida de la capa anterior en un vector unidimensional antes de pasarla a capas completamente conectadas.
7. Dense: Una capa densa con 256 neuronas y una función de activación ReLU.
8. Dense: Otra capa densa con 128 neuronas y una función de activación ReLU.
9. Dense: La capa de salida, con un número de neuronas igual a numClass, que utiliza una función de activación softmax.

# 4. Resultados

# Después de completar el entrenamiento de los modelos, hemos obtenido resultados significativos que son esenciales para evaluar la efectividad de las estrategias empleadas en la detección de felinos en imágenes capturadas por cámaras de foto trampa. Para una comparación detallada de los resultados, hemos utilizado curvas de aprendizaje, representando la precisión (accuracy) en el eje y y las épocas de validación en el eje x. Estos resultados se presentan en la ilustrativa Figura 4.1.

# 

Figura 4.1: Resultados de comparativas con aumentación de datos vs. sin aumentación de datos

Ahora, profundicemos en el análisis específico de los resultados obtenidos para cada modelo:

**Modelo sin aumentación de datos (Figura 4.2)**

Mediante el análisis de la tabla de resultados de épocas de validación sin aumentación de datos (Figura 4.2), se revela que desde el inicio del entrenamiento, los valores de precisión se mantienen consistentes, permaneciendo en un rango relativamente estrecho. Este fenómeno plantea un punto de interés, ya que sugiere que el modelo podría estar experimentando overfitting desde las primeras etapas de entrenamiento. Esta observación refuerza la necesidad de implementar una estrategia más efectiva en el proceso de entrenamiento para abordar este problema.

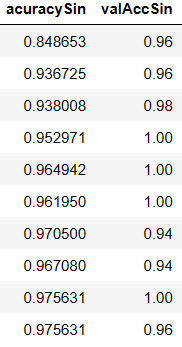


Figura 4.2: Tabla de resultados épocas validación sin aumentación de datos.

Es esencial señalar que el modelo sin aumento de datos no logra un aumento significativo en la precisión a lo largo de las épocas de validación, lo que sugiere un estancamiento en su capacidad de generalización.

**Modelo con aumentación de datos (Figura 4.3).**

Al observar los resultados en la tabla de épocas de validación con aumentación de datos (Figura 4.3), se observa un comportamiento diferente. Desde la primera época, la precisión alcanza un valor inicial del 73.6% y experimenta un aumento gradual sin evidencia de overfitting. Este modelo muestra una tendencia positiva y concluye con una precisión destacable del 91.6%.

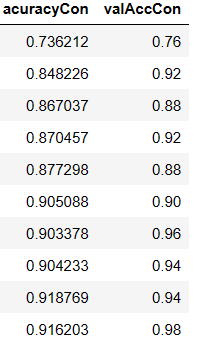


Figura 4.3: Tabla de resultados épocas validación con aumentación de datos.

Es interesante notar que el modelo con aumentación de datos demuestra una capacidad de adaptación más efectiva y gradual, evidenciando su capacidad para aprender patrones complejos de manera más precisa y generalizable.

**Análisis y Evaluación de Parámetros.**

Los resultados obtenidos plantean diversas consideraciones sobre qué parámetros influyen en el rendimiento de los modelos. El análisis revela la importancia crítica de la aumentación de datos en la mejora de la precisión del modelo. La técnica de aumentación de datos, al introducir variaciones controladas en las imágenes de entrenamiento, amplía la diversidad de datos disponibles para el modelo, ayudando a prevenir el overfitting y mejorando la capacidad de generalización.

En cuanto a la arquitectura de la red neuronal (Figura 3.2), la disposición de las capas Convolucionales (Conv2D) y las capas completamente conectadas (Dense) fue elegida para aprovechar la capacidad de las redes neuronales convolucionales en la detección de patrones en imágenes. La elección de las funciones de activación ReLU en las capas Convolucionales y Dense contribuye a la eficiencia en el entrenamiento y mejora la no linealidad del modelo.

# 5. Conclusión

En conclusión, el desarrollo de un algoritmo de clasificación de felinos en imágenes mediante técnicas de machine learning y computer vision ha demostrado ser un paso significativo hacia la preservación y estudio de la fauna en reservas naturales. La implementación de redes neuronales convolucionales (CNN) ha sido fundamental para alcanzar el objetivo de detectar y clasificar diferentes especies de felinos con una eficacia no menor al 85%.

Tras el análisis de los resultados obtenidos, se evidencia que el modelo que incorpora aumentación de datos ha destacado como la opción más viable. La utilización de rotación horizontal y vertical, zoom y rotación de imágenes ha permitido un entrenamiento más robusto, evitando el sobreajuste y obteniendo un elevado nivel de precisión, alcanzando un 91.6% de accuracy. Esta estrategia ha mejorado la generalización del modelo y, en consecuencia, su capacidad para reconocer nuevas instancias de felinos en imágenes capturadas en estado salvaje.

El enfoque de computer vision y deep learning, mediante las redes neuronales convolucionales, ha brindado resultados prometedores y sienta las bases para futuros desarrollos en el campo de la conservación de la vida silvestre. La capacidad de detectar felinos en fotografías automáticamente permitirá a los investigadores de reservas naturales enfocarse en analizar y comprender mejor las poblaciones de estas especies, proporcionando una herramienta valiosa para la protección y preservación de la biodiversidad.

No obstante, a pesar de los logros alcanzados en este trabajo, es importante mencionar que siempre hay margen de mejora y oportunidades para perfeccionar el algoritmo. La incorporación de más datos etiquetados y variados, así como la experimentación con diferentes arquitecturas de redes neuronales, pueden enriquecer aún más la precisión y robustez del modelo.

En resumen, el desarrollo de este algoritmo ha sido un paso significativo hacia la automatización y optimización de la clasificación de felinos en imágenes, contribuyendo al conocimiento y protección de estas especies en peligro de extinción. La combinación de la visión por computadora y el aprendizaje profundo ha demostrado ser una poderosa herramienta en la conservación y estudio de la vida silvestre, y su potencial es prometedor para futuras aplicaciones en la preservación del medio ambiente y la biodiversidad.

# 6. Bibliografía

1. Gerry. (2023). *10 Big Cats of the Wild - Image Classification*. Kaggle.com. <https://www.kaggle.com/datasets/gpiosenka/cats-in-the-wild-image-classification>
2. *What is Computer Vision? | IBM*. (2022). Ibm.com. <https://www.ibm.com/topics/computer-vision#what-is-computer-vision->
3. López-Saca, F. (2019). Clasificación de imágenes usando redes neuronales convolucionales. *Azc.uam.mx*. <http://hdl.handle.net/11191/6123>
4. *¿Qué es deep learning?* (2017). Sas.com. <https://www.sas.com/es_ar/insights/analytics/deep-learning.html>
5. *¿Qué es una red neuronal convolucional? | 3 cosas que debe saber*. (2023). Mathworks.com. <https://es.mathworks.com/discovery/convolutional-neural-network-matlab.html#:~:text=Una%20CNN%20consta%20de%20una,activaci%C3%B3n%20o%20ReLU%2C%20y%20agrupaci%C3%B3n.>
6. Adegoke, K. (2018). *A new Fibonacci identity and its associated summation identities*. ArXiv.org. https://arxiv.org/abs/1809.06850
7. ‌

‌

‌