

# Desarrollo de un modelo de Deep Learning para el conteo y detección de animales en manadas densas a partir de imágenes aéreas

Víctor Pérez<sup>1</sup>, Jordi Sánchez<sup>2</sup>, Simón Aristizábal<sup>3</sup>, Maryi Carvajal<sup>4</sup>

*Maestría en Inteligencia Artificial,*

*Universidad de los Andes,*

*Bogotá, Colombia*

**Abstract**—Este trabajo presenta una extensión de la arquitectura HerdNet para la detección y conteo multiespecie de fauna en imágenes aéreas, mediante la incorporación del módulo de atención CBAM (Convolutional Block Attention Module) y el uso de la técnica de Hard Negative Patching (HNP) en un esquema de entrenamiento en dos fases. A pesar de que el modelo HerdNet+CBAM no superó los resultados reportados en estudios previos, sí logró mejorar el rendimiento frente al modelo base implementado en este estudio, especialmente en términos de precisión por clase y estabilidad durante la validación. Asimismo, la aplicación de HNP contribuyó a reducir falsos positivos y a mejorar la discriminación entre regiones ambiguas del fondo y las verdaderas instancias de animales. Estos hallazgos posicionan a HerdNet+CBAM como una alternativa viable y prometedora para el análisis automático de fauna silvestre en entornos complejos, destacando la importancia de ajustar los mecanismos de atención al dominio y de complementar las arquitecturas con estrategias de entrenamiento robustas. El presente estudio constituye la primera implementación documentada de HerdNet combinada con CBAM en este dominio, ofreciendo una base sólida para investigaciones futuras en sistemas de monitoreo ecológico mediante visión por computador.

**Index Terms**—detección de objetos, conteo de animales, HerdNet, aprendizaje profundo, imágenes aéreas, CBAM, DLA

## I. INTRODUCCIÓN

La detección y cuantificación de animales en manadas densas a partir de imágenes aéreas representa un desafío significativo para los sistemas automatizados de visión por computador, debido a factores como occlusiones, distribuciones irregulares, variaciones en escala, y ambientes con ruido visual [1]. HerdNet, una arquitectura convolucional de una sola etapa, ha demostrado un rendimiento competitivo en escenarios de alta densidad al combinar mecanismos de localización puntual y clasificación por especie. Esta arquitectura está diseñada para producir mapas de calor que indican la presencia de individuos dentro de una imagen, lo cual permite una localización precisa incluso en condiciones de alta superposición entre objetos [4].

HerdNet integra un codificador (encoder) basado en DLA-34, que extrae características jerárquicas conservando la información espacial, un cuello de botella (“bottleneck”) para

sintetizar representaciones discriminativas, y dos ramas de salida: una para la detección mediante mapas de calor y otra para la clasificación por especie a partir de los embeddings generados [4]. Esta configuración ha mostrado eficacia en tareas de monitoreo de fauna, sin embargo, enfrenta limitaciones frente a la complejidad visual de entornos naturales, especialmente en la diferenciación entre regiones objetivo y fondo.

Para mejorar la capacidad de los modelos convolucionales en la identificación de regiones relevantes dentro de imágenes complejas, se propone tres alternativas diferentes desde una reproducción de HerdNet con un early stopping, una modificación en la capa DLA con 60 capas y por último, el uso del Convolutional Block Attention Module (CBAM), un módulo de atención ligero y eficiente que ha demostrado mejorar el desempeño en tareas de detección cuando se integra a arquitecturas existentes, como es el caso de YOLOv5 [7]. Inspirados por estos resultados, en este trabajo se propone una extensión de HerdNet que incorpora CBAM justo después del cuello de botella y antes del decodificador. Esta integración busca reforzar la diferenciación entre el objetivo (animales) y el fondo en escenas aéreas de alta densidad, aprovechando los mecanismos secuenciales de atención por canal y espacial que ofrece CBAM para mejorar la precisión sin incurrir en un costo computacional significativo.

CBAM actúa como un refinador de características intermedias, permitiendo a la red aprender “qué” y “dónde” atender. Al aplicar atención primero a nivel de canales y luego espacialmente, este módulo potencia las regiones discriminativas y suprime activaciones irrelevantes. La motivación detrás de esta adición es permitir que la red concentre sus recursos computacionales en las zonas más informativas de la imagen, minimizando errores derivados del ruido de fondo, occlusiones o patrones engañosos presentes en el entorno natural.

Los beneficios de integrar CBAM en tareas de detección incluyen:

- **Mejor discriminación entre clases:** al recalibrar la importancia relativa de cada canal de características, la red puede enfocarse en patrones morfológicos y texturales distintivos de cada especie.

<sup>1</sup>v.perezd@uniandes.edu.co

<sup>2</sup>jn.sanchez1@uniandes.edu.co

<sup>3</sup>simon.aristizabal112345@uniandes.edu.co

<sup>4</sup>ma.carvajal2@uniandes.edu.co

- **Mayor robustez a oclusiones y ruido:** la atención espacial permite priorizar regiones compactas donde se concentran información relevante, mitigando el impacto de regiones irrelevantes o ruidosas.
- **Refuerzo semántico-local:** CBAM permite a la red mantener información contextual sin necesidad de complementar la arquitectura con capas profundas adicionales.
- **Compatibilidad con arquitecturas existentes:** su bajo costo computacional y facilidad de integración hacen de CBAM una extensión versátil para mejorar modelos sin comprometer el rendimiento.

De este modo, se espera que HerdNet+CBAM no solo conserve las ventajas de su diseño original, sino que amplifique su capacidad de generalización y detección precisa en entornos naturales complejos, donde factores como camuflaje, iluminación variable y solapamiento entre individuos afectan negativamente el rendimiento de modelos convencionales.

Este artículo se divide en cinco partes principales: una descripción de la arquitectura propuesta con la integración de CBAM, una explicación detallada de la metodología empleada, la presentación de los resultados obtenidos y, finalmente, las conclusiones del estudio.

## II. ESTADO DEL ARTE

La detección automatizada de fauna silvestre en imágenes aéreas ha ganado gran relevancia en la última década, especialmente con el uso de plataformas UAV (vehículos aéreos no tripulados) y el auge de los modelos de aprendizaje profundo. Entre los enfoques más eficaces en este dominio se encuentran los métodos basados en redes convolucionales profundas (CNN) [10], los cuales han demostrado una alta capacidad para adaptarse a la complejidad visual de ambientes naturales, donde los objetos de interés (animales) presentan oclusiones, variaciones de escala, camuflaje y distribución densa.

Uno de los estudios más relevantes es el de Delplanque et al. [1], quienes emplearon CNNs para la identificación multi-especie de mamíferos africanos en imágenes UAV, demostrando que modelos convolucionales bien calibrados pueden alcanzar precisiones competitivas incluso en condiciones de campo. En particular, su trabajo remarcó la importancia de arquitecturas especializadas para escenas con alta densidad animal, ya que modelos genéricos como Faster R-CNN o YOLO suelen presentar un rendimiento subóptimo cuando la distancia entre objetos es mínima o existe alta superposición [9].

En este contexto, la arquitectura HerdNet se posiciona como una solución eficaz para la tarea específica de conteo y clasificación puntual de animales en manadas densas. A diferencia de métodos tradicionales de segmentación o detección por cajas delimitadoras, HerdNet adopta una aproximación basada en mapas de calor de localización y mapas de clasificación en paralelo, lo que permite una predicción precisa en el centroide de cada individuo sin requerir la segmentación de su silueta completa. Esta aproximación resulta altamente escalable y más robusta frente a la oclusión parcial, tal como se refleja en los resultados obtenidos en este trabajo base.

La estructura del modelo combina un backbone DLA-34 (Deep Layer Aggregation) como codificador, optimizado para capturar características multi-escala de manera eficiente, con un decodificador ligero que genera salidas densas para localización y clasificación. Esta arquitectura se alinea con las recomendaciones planteadas por Kellenberger et al. [5], quienes subrayan la necesidad de modelos livianos y rápidos para ser implementados en flujos de trabajo operativos sobre grandes extensiones de datos UAV.

Sin embargo, a pesar de su diseño especializado, el modelo base aún enfrenta limitaciones asociadas a la confusión entre especies morfológicamente similares, como se evidencia en la matriz de confusión obtenida, donde ocurren cruces frecuentes entre las clases *kob*, *topi* y *buffalo*. Esta problemática ha sido documentada también en trabajos recientes como el de Delplanque et al. (2023) [4], donde se propone un enfoque más preciso de conteo y clasificación mediante aprendizaje profundo, incorporando mecanismos de validación cruzada visual y muestreo adaptativo.

Adicionalmente, este tipo de bases de datos presentan un alto desbalanceo entre las clases de predecir, problemática que ha sido abordada por otros autores como B. Kellenberger et al. [11] a través de pesos en la función de perdida.

Cabe resaltar que, aunque HerdNet no incorpora mecanismos explícitos de atención o auto-pesado de canales como en CBAM o SE blocks [2], sus resultados iniciales en términos de recall y precisión por clase son competitivos. Esta base sólida justifica la posterior incorporación de módulos de atención como mejora arquitectónica, y posiciona al modelo como una referencia válida dentro del estado del arte para detección densa de fauna en contextos reales.

Finalmente, la detección y conteo multiespecie de fauna en imágenes aéreas es de gran relevancia, ya que, aporta datos fundamentales para entender la estructura de comunidades biológicas, relaciones entre especies (depredación, competencia, simbiosis), salud ecológica general o implementación de agricultura sostenible [12], [13].

## III. ARQUITECTURA HERDNET+CBAM

La arquitectura propuesta mantiene la base de HerdNet, que consta de un codificador (*encoder*) basado en la red DLA-34, un cuello de botella que actúa como extractor de características globales, y dos cabezas: una para generar mapas de calor asociados a la localización puntual de instancias animales, y otra dedicada a la clasificación por especie a partir de características semánticas.

La principal modificación introducida es la inserción de un módulo CBAM al final del bottleneck, antes del decodificador. CBAM consiste en dos submódulos: (1) el **módulo de atención por canal**, que genera una ponderación adaptativa de cada canal de características en función de estadísticas agregadas como promedio y máximo, y (2) el **módulo de atención espacial**, que aplica una convolución sobre mapas agregados para recalibrar espacialmente la importancia de cada región en la imagen (ver figura 2).

La arquitectura base de HerdNet se compone de tres módulos principales:

- 1) Un **encoder DLA-34**, que combina representaciones a diferentes escalas con conexiones jerárquicas para preservar información espacial mientras se aumenta la profundidad semántica.
- 2) Un **cuello de botella (bottleneck)** que sintetiza las características extraídas en un espacio latente compartido.
- 3) Dos **cabezas de salida**: una para regresión de mapas de calor (detección puntual) y otra para clasificación por especie basada en embeddings extraídos.

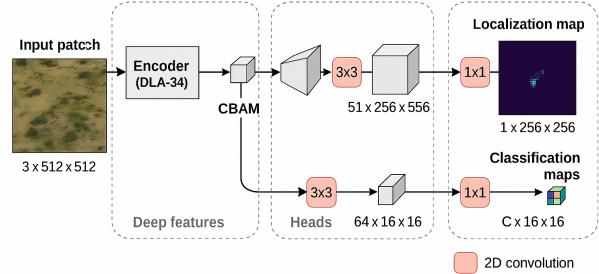


Fig. 2: Arquitectura con modelo CBAM

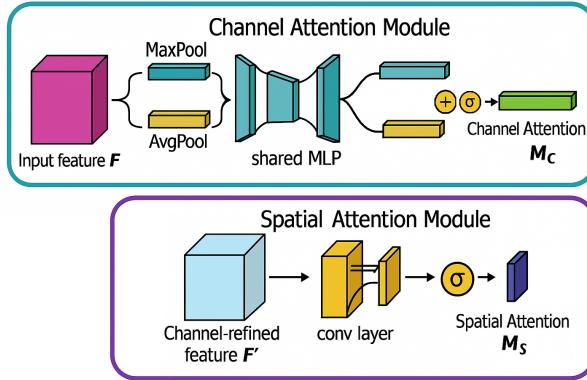


Fig. 1: Arquitectura de CBAM, arriba se encuentra el canal del módulo de atención y en la parte inferior el módulo espacial de atención.

Sobre esta arquitectura se incorpora el módulo CBAM inmediatamente al finalizar el bottleneck (ver fig: 2). CBAM consta de dos bloques secuenciales:

- **Módulo de atención por canal:** modela la importancia de cada canal utilizando operaciones globales de average pooling y max pooling, seguidas de una red neuronal completamente conectada para generar un vector de pesos por canal.
- **Módulo de atención espacial:** opera sobre la salida anterior, concatenando las proyecciones promedio y máxima a lo largo de los canales, y aplicando una convolución de tamaño fijo para generar un mapa de atención espacial.

El bloque CBAM introduce aproximadamente 32.800 parámetros adicionales, lo que representa un aumento marginal considerando los beneficios en representación y especialización de características. Además, su posición antes del decodificador asegura que las mejoras de atención impacten directamente la calidad de los mapas de salida (ver figura 2).

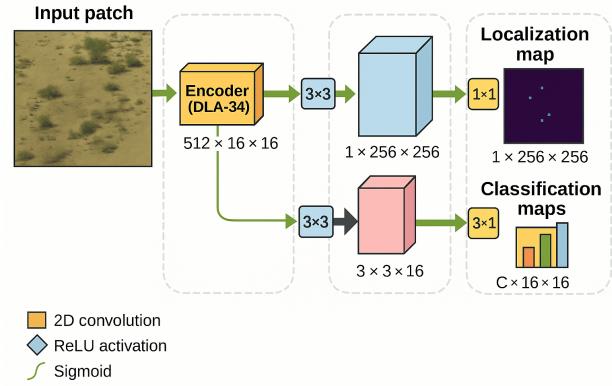


Fig. 3: Arquitectura base Herdnet

Esta modificación mantiene el resto de la arquitectura HerdNet sin alteraciones, lo cual facilita la comparación controlada entre la versión original y la extendida con CBAM, permitiendo medir de forma aislada el impacto del mecanismo de atención sobre las tareas de localización y clasificación de fauna.

#### IV. METODOLOGÍA

La experimentación se llevó a cabo con el objetivo de evaluar el impacto de introducir el módulo CBAM en la arquitectura HerdNet, comparando su rendimiento con versiones previas del modelo sin atención y con modificaciones estructurales distintas, tales como la implementación de DLA 60. Este análisis busca determinar si la incorporación de mecanismos de atención espacial y por canal puede incrementar la sensibilidad y robustez del modelo frente a la variabilidad del entorno y a la complejidad inherente en las imágenes aéreas de fauna.

##### A. Dataset y Preprocesamiento

Se utilizó el dataset Ennedi, presentado por Delplanque et al. [1], el cual contiene miles de imágenes aéreas de fauna africana, cada una etiquetada con las coordenadas y especies

identificadas. Este conjunto de datos representa un entorno realista caracterizado por alta densidad de objetos, solapamientos frecuentes y fondos con textura natural compleja.

Este conjunto contiene anotaciones de seis especies animales (Buffalo, Elephant, Kob, Topi, Warthog y Waterbuck) observadas en imágenes aéreas. Las instancias se dividen en subconjuntos de entrenamiento (62–73%), validación (7–16%) y prueba (15–25%) según la especie, con un total de 10,239 ejemplos. La distribución entre clases es desigual, siendo Elephant y Warthog las menos representadas, y Waterbuck la más frecuente, lo cual plantea un desafío adicional de desbalance de clases para las tareas de detección y clasificación. Para abordar esta problemática se decidió agregar un peso a la función de perdida por tipo de especie, permitiendo penalizar fuertemente cuando el modelo se equivoque en categorías menos representadas (ver TABLE I).

TABLE I: Distribución de clases y pesos asignados por especie

Especie	Topi	Buffalo	Kob	Elephant	Warthog	Waterbuck
Total	1678	1058	1732	316	166	2012
Peso	1.0	2.0	1.0	6.0	12.0	1.0

Las imágenes fueron segmentadas en parches de 512x512 px mediante la herramienta `tools/patcher.py`, con un traslape de 160, tal como es realizado por A. Delplanque et al. [4]. La columna `species` fue agregada manualmente a los archivos CSV de etiquetas para permitir la clasificación multiclas. Las imágenes fueron normalizadas a valores entre 0 y 1, y se empleó aumento de datos (*data augmentation*) durante el entrenamiento: rotaciones aleatorias (0-30 grados), cambios de brillo y contraste, recortes aleatorios (*cutout*) y perturbaciones gaussianas, todo implementado con `Albumentations`.

#### B. Configuración del Entorno y Hiperparámetros

El entrenamiento fue realizado en una VM con GPU Nvidia A40-24C de 24 GB, ejecutando Ubuntu 20.04 y PyTorch 1.13.1. Se configuró un entorno virtual con las dependencias especificadas en el archivo `requirements.txt` del repositorio original. Se utilizó Python 3.8 para garantizar compatibilidad con la versión base del proyecto.

Los hiperparámetros seleccionados fueron:

- **Batch size:** 16
- **Epochs:** 100
- **Optimizer:** AdamW con weight decay de  $1 \times 10^{-5}$
- **Learning rate:**  $3 \times 10^{-4}$  con scheduler ReduceLROnPlateau
- **Early stopping:** activado tras 10 epochs sin mejora en F1-score
- **Loss functions:** Focal Loss para localización; Cross-Entropy ponderada para clasificación

#### C. Diseño Experimental

El experimento principal consistió en entrenar tres versiones del modelo HerdNet:

- 1) **Modelo base:** HerdNet sin cambios, con DLA-34 como backbone
- 2) **Modelo propuesto 1:** Herdnet sin cambios, usando DLA-60 como backbone
- 3) **Modelo propuesto 2:** HerdNet + CBAM, con el módulo de atención aplicado a la salida del bottleneck

Para cada versión se registraron las métricas de F1-score, precisión y recall, tanto en el conjunto de validación como en el conjunto de prueba. Adicionalmente, se empleó la plataforma `wandb` para el seguimiento de curvas de aprendizaje y comparación visual del comportamiento durante el entrenamiento.

#### D. Implementación con DLA 60

Este experimento parte de la modificación del *backbone* que utiliza el modelo DLA-34. Se modificó el número de capas, pasando de 34 a 60 capas entrenables. Esto con el fin de que el modelo tenga la capacidad de identificar características o patrones más complejos o detallados en las imágenes. Para esto se utilizó un modelo preentrenado sobre el dataset de ImageNet (Modelo DLA de 60 Capas). El resto del modelo HerdNet se mantuvo sin cambios en este experimento.

```
Herdnet = HerdNet(num_layers=60,
num_classes=num_classes,
down_ratio=down_ratio).cuda()
```

#### E. Implementación de CBAM

El módulo CBAM utilizado está compuesto por dos bloques: atención por canal y atención espacial. Ambos bloques fueron configurados con:

- `gate_channels`: 512
- `reduction_ratio`: 16
- `spatial_kernel`: 7
- `use_residual`: False (para evitar introducción de ruido acumulado)

Este módulo fue integrado en el archivo `models/herdnet.py` y llamado dentro de la clase principal justo después de la salida del bottleneck:

```
self.cbam = cbam_modules.CBAM(
    gate_channels=512,
    reduction_ratio=16,
    spatial_kernel=7,
    use_residual=False)
```

```
bottleneck = self.cbam(bottleneck)
```

La clase CBAM fue desarrollada siguiendo la propuesta de Woo et al. (2018). Se optó por una implementación modular, que permite activar o desactivar CBAM mediante el archivo de configuración YAML.

#### F. Segundo entrenamiento con Hard Negative Patching (HNP)

Tras entrenar el modelo durante 100 épocas, se procedió a una segunda fase de entrenamiento enfocada en refinar su capacidad de discriminación mediante la técnica de *Hard*

*Negative Patching* (HNP). Esta técnica consiste en identificar ejemplos negativos difíciles (por ejemplo, zonas del fondo con texturas o formas similares a animales reales) y reutilizarlos como muestras adicionales para reentrenar el modelo. Su uso es ampliamente reconocido por su impacto en la mejora de la precisión de clasificadores en contextos donde el fondo complejo o la alta densidad de objetos genera ambigüedades semánticas [6].

En este segundo ciclo de entrenamiento, se reutilizó la arquitectura previamente entrenada, incluyendo el encoder DLA-34 y el bloque CBAM, y se introdujeron imágenes seleccionadas manualmente que contenían principalmente zonas sin animales, pero con alto potencial de inducir falsos positivos. Estas imágenes se incorporaron como parches negativos difíciles, siguiendo una estrategia de balance entre clases minoritarias y zonas sin objeto para reforzar el aprendizaje supervisado en las regiones más confusas del dominio visual.

El entrenamiento se llevó a cabo durante 50 épocas, utilizando un tamaño de *batch* de 4 imágenes, lo cual permitió una mejor adaptación al límite de memoria de la GPU sin sacrificar estabilidad en el gradiente. Como resultado, el modelo mostró una reducción significativa en la tasa de falsos positivos, especialmente en clases con alto solapamiento visual, como *kob*, *topi* y *buffalo*, manteniendo al mismo tiempo un buen rendimiento en especies minoritarias como *warthog* y *elephant*. Esta mejora se alinea con lo reportado por Shrivastava et al. [6], quienes demostraron que el *Online Hard Example Mining* (OHEM), técnica relacionada al HNP, incrementa sustancialmente la precisión de los detectores en escenarios densos.

La utilidad del HNP se justifica aún más si se considera la naturaleza del dataset utilizado, compuesto por imágenes aéreas de fauna africana tomadas desde drones. Como se ha señalado en estudios previos [1], este tipo de imágenes presenta desafíos únicos debido a la oclusión parcial, la variación de escala y la similitud cromática entre animales y el entorno. En este contexto, la selección cuidadosa de parches negativos permite reforzar la capacidad del modelo para ignorar artefactos visuales irrelevantes, y enfocar el aprendizaje en regiones discriminativas. Además, investigaciones recientes como la de Delplanque et al. [4] han propuesto enfoques similares, con resultados prometedores en la detección y conteo de animales desde imágenes UAV, destacando el valor de estrategias avanzadas de entrenamiento como el HNP para mejorar la generalización en escenarios reales.

En síntesis, el segundo entrenamiento con HNP representó un paso crucial para mejorar la precisión del modelo en condiciones más cercanas a su despliegue operativo, reduciendo la ambigüedad del fondo y fortaleciendo su capacidad de discriminación espacial sin comprometer la sensibilidad frente a clases minoritarias.

#### G. Evaluación y Validación

El modelo fue evaluado en el conjunto de prueba realizando la medida de cada métrica a partir de la detección de puntos, para lo que se aplica un radio como umbral de distancia para

determinar si una predicción es un verdadero positivo. En este caso, se utilizó el radio de 5 por defecto para cada evaluación y se determinaron las siguientes métricas:

- **Precision:** proporción de predicciones positivas correctas.
- **Recall:** proporción de verdaderos positivos detectados.
- **F1-Score:** media armónica entre precisión y recall.
- **Confusion matrix:** por clase, para identificar errores sistemáticos.

La matriz de confusión fue utilizada para analizar la tasa de confusión interclase, y para identificar posibles mejoras en la asignación de pesos en la función de pérdida. Se estudió el comportamiento del modelo por especie y se analizaron visualmente los mapas de activación producidos por CBAM para validar su interpretabilidad.

## V. RESULTADOS Y DISCUSIÓN

La presentación de resultados se dividirá en tres partes: la reproducción del artículo principal implementando early stopping, el uso de DLA 60 y por último, HerdNet+CBAM. Cada subsección presentará métricas de evaluación y entrenamiento.

### A. Modelo HerdNet Base Early Stopping

El modelo base de HerdNet se entrenó con una leve variación que es realizar un early Stopping, esto para evitar un leve overfitting que se observó en la etapa de experimentación. El entrenamiento fue realizado en dos etapas, primero se entrenó el modelo durante treinta épocas, luego se utilizó la técnica de HNP, tal como se describe en el artículo original de HerdNet. Con esto se obtuvo un rendimiento inferior al presentado por A. Delplanque et al [4]. Lo que se puede deber a una posible underfitting causada por la disminución de épocas (ver Tabla II).

TABLE II: Resumen de métricas del modelo base por categoría

Clase	Recall	Precision	F1_Score	Confusion
Topi	0.8489	0.7346	0.7876	0.0890
Buffalo	0.8567	0.4643	0.6022	0.0685
Kob	0.8218	0.7382	0.7778	0.1765
Elephant	0.5676	0.0959	0.1641	0.1064
Warthog	0.6111	0.1594	0.2529	0.3714
Waterbuck	0.8052	0.1140	0.1997	0.0036

La Figura 4 muestra la matriz de confusión del modelo base. Se evidencia una confusión significativa de la clase *Kob* hacia *Topi*, así como errores de clasificación entre *Buffalo* y *Elephant*, indicando posibles limitaciones en la diferenciación de estas especies bajo ciertas condiciones visuales.

Confusion Matrix							
True Label	Topi	570	2	22	9	4	0
	Buffalo	0	284	0	18	4	0
	Kob	87	0	386	0	0	0
	Elephant	0	1	0	49	0	0
	Warthog	2	1	0	1	29	0
	Waterbuck	0	4	0	0	0	554
	Predicted Label	Topi	Buffalo	Kob	Elephant	Warthog	Waterbuck

Fig. 4: Matriz de confusión del modelo base usando datos de test

#### B. Modelo HerdNet backbone DLA60

Con el fin de aumentar la capacidad del modelo para capturar patrones más complejos, se modificó el *backbone* original DLA-34 por una versión de 60 capas (DLA60).

La Tabla 3 presenta los resultados de este modelo. Si bien se observan mejoras puntuales, como el aumento en el *recall* para *Waterbuck* (0.8852), otras clases como *Elephant* y *Buffalo* mostraron caídas notables en la precisión (0.0791 y 0.3555, respectivamente). En términos generales, el modelo no logró superar al modelo base en métricas globales, sugiriendo que la complejidad añadida podría haber derivado en *overfitting*.

TABLE III: Resumen de métricas del modelo DLA60 por categoría

Clase	Recall	Precision	F1_Score	Confusion
Topi	0.8237	0.6304	0.7142	0.1299
Buffalo	0.7106	0.3555	0.4737	0.2369
Kob	0.7715	0.7683	0.7699	0.2285
Elephant	0.5676	0.0791	0.1388	0.0233
Warthog	0.5000	0.2903	0.3673	0.5000
Waterbuck	0.8852	0.0998	0.1794	0.0000

#### C. Modelo HerdNet + CBAM

TABLE IV: Resumen de métricas del modelo CBAM por categoría

Clase	Recall	Precision	F1_Score	Confusion
Topi	0.8474	0.7637	0.8034	0.0419
Buffalo	0.5931	0.7870	0.6765	0.1375
Kob	0.8574	0.8279	0.8424	0.1316
Elephant	0.2973	0.2222	0.2543	0.0435
Warthog	0.7222	0.4063	0.5200	0.0714
Waterbuck	0.6701	0.3842	0.4884	0.0000

Confusion Matrix							
True Label	Topi	578	0	7	8	10	0
	Buffalo	0	224	0	30	4	1
	Kob	59	0	409	1	2	0
	Elephant	0	0	0	25	1	1
	Warthog	1	0	1	0	27	0
	Waterbuck	0	0	0	0	0	488
	Predicted Label	Topi	Buffalo	Kob	Elephant	Warthog	Waterbuck

Fig. 5: Matriz de confusión del modelo de HerdNet+CBAM

En cuanto a la matriz de confusión (Fig. 5) y las métricas por clase (Tab. IV), se observa que el modelo HerdNet+CBAM logra buenos niveles de desempeño en especies como *Kob* y *Topi*, alcanzando F1-scores de 0.84 y 0.80 respectivamente, lo que sugiere una sólida discriminación en clases con suficiente representación y morfología distintiva.



Fig. 6: Ejemplo visual del modelo HerdNet+CBAM sobre una imagen aérea. Las etiquetas numéricas indican instancias detectadas, y se incluye un recuadro con ampliación para resaltar la detección de un animal parcialmente oculto

La Figura 6 muestra un ejemplo visual del resultado del modelo HerdNet+CBAM sobre una imagen aérea, con etiquetas generadas para múltiples instancias animales. Se destaca además una zona ampliada que permite visualizar con mayor claridad la detección de un individuo parcialmente oculto entre la vegetación. Esta imagen ilustra las dificultades del dominio (camuflaje, sombra, escala) y la capacidad del modelo para discriminar entre fondo y objeto, incluso en condiciones adversas.

A nivel global (Tabla 5), el modelo CBAM obtiene un F1 de 0.704 en test y una precisión de 0.634, con un *recall* de (0.792), lo que indica una tendencia a generar más verdaderos positivos a costa de una menor precisión.

Sin embargo, al observar la Tabla 6, que resume las métricas principales de los tres modelos evaluados, se concluye que

el modelo CBAM no supera al modelo presentado por A. Delplanque et al [4]. Su rendimiento en el conjunto de prueba es inferior en F1 (0.358 vs. 0.430) y precisión (0.2280 vs. 0.3844), indicando una generalización más débil. A pesar de contar con mayor capacidad y mayor número de épocas de entrenamiento, el modelo CBAM sugiere una tendencia al sobreajuste al conjunto de entrenamiento, lo que podría deberse a una introducción de ruido en los mecanismos de atención.

Aunque el modelo con CBAM no superó las métricas reportadas por A. Delplanque et al. [4], sí logró mejorar el desempeño respecto al modelo base desarrollado en este trabajo (ver Tabla V). Este resultado posiciona a HerdNet+CBAM como una alternativa prometedora, especialmente si se dispone de un conjunto de datos más amplio que contribuya a mitigar el desbalance entre clases. El objetivo es permitir que el modelo aprenda eficazmente sobre imágenes aéreas de fauna silvestre, una tarea que, hasta donde se tiene conocimiento, no ha sido abordada previamente en la literatura utilizando esta combinación de arquitecturas.

Una posible hipótesis es que las características visuales en el dominio de imágenes aéreas de fauna (con fondos altamente texturizados, camuflaje y solapamiento entre clases) no permiten a CBAM generar mapas de atención suficientemente discriminativos. Al ser una arquitectura inicialmente diseñada para escenarios más controlados (por ejemplo, detección en objetos sobre fondo uniforme), su aplicación directa sin ajustes específicos podría introducir ruido adicional al paso de inferencia, en lugar de reducirlo.

TABLE V: Resumen de métricas principales de distintos experimentos

Experimento	Epochs	F1 Val	F1 Test	Precision Test	Recall Test
Base	70	0.820	0.520	0.367	0.882
DLA60	55	0.819	0.390	0.2433	0.9261
CBAM	150	0.825	0.704	0.634	0.792

## VI. CONCLUSIONES

Este trabajo presenta una extensión de la arquitectura HerdNet mediante la incorporación del módulo de atención CBAM, con el objetivo de mejorar la detección multiespecie en imágenes aéreas de fauna silvestre. Si bien HerdNet+CBAM no superó los resultados obtenidos por trabajos previos como el de A. Delplanque et al. [4], mostró mejoras con respecto al modelo base implementado en este estudio, lo que evidencia su potencial como arquitectura complementaria para escenarios de alta densidad (Tabla V).

Estos hallazgos positionan a HerdNet+CBAM como una alternativa prometedora, especialmente considerando su capacidad para beneficiarse de un conjunto de datos más amplio que ayude a mitigar el desbalance entre clases. Además, este trabajo representa una contribución relevante a la literatura, al aplicar por primera vez esta combinación de arquitecturas para el análisis de imágenes aéreas de fauna silvestre. Esto abre nuevas posibilidades para avanzar en sistemas automáticos de

monitoreo ecológico, ofreciendo una base sólida para investigaciones futuras en contextos de biodiversidad y conservación.

Aunque la arquitectura HerdNet+CBAM mostró un desempeño durante la fase de validación de  $F1 = 0.825$ , su rendimiento en el conjunto de prueba fue de  $F1 \text{ test} = 0.704$ . Este comportamiento sugiere la presencia de sobreajuste, posiblemente debido a la limitada cantidad de datos disponibles y al desbalance entre clases. La inclusión del módulo CBAM supuso un aumento de aproximadamente 32,800 parámetros, equivalente a una fracción mínima respecto al total aproximado de 18 millones de parámetros que conforman la arquitectura base de HerdNet. Sin embargo, este aumento no fue suficiente para generar una mejora sistemática en el desempeño del modelo, lo que indica que el desafío no radica únicamente en la capacidad del modelo, sino en la manera en que dicha capacidad se emplea para capturar patrones relevantes dentro del dominio específico de imágenes aéreas de fauna silvestre.

Se debe destacar que los resultados anteriores sugieren que HerdNet, incluso en su configuración base, ya es eficaz para capturar la semántica necesaria para la detección puntual en contextos densos, y que la introducción de CBAM podría haber interferido con esta capacidad si no es calibrada cuidadosamente. Esto abre preguntas sobre la necesidad de mecanismos de atención adaptativos al contexto, en lugar de mecanismos genéricos, y destaca la importancia de realizar estudios de sensibilidad para evaluar su impacto antes de su integración definitiva.

Es importante enfatizar qué en entornos como el monitoreo aéreo de fauna salvaje, no siempre la incorporación de arquitecturas más complejas garantiza un mejor desempeño. El balance entre simplicidad, eficiencia computacional y generalización debe guiar el diseño de futuras mejoras arquitectónicas. A futuro, se recomienda investigar variantes de atención específicas para detección densa en entornos naturales, evaluar mecanismos alternativos como los bloques SE o mecanismos auto-regresivos, y continuar explorando ajustes finos de loss, estrategias de regularización, y configuraciones de entrenamiento con *hard examples*.

En este sentido, una de las estrategias más relevantes implementadas en este trabajo fue el entrenamiento en dos etapas con la técnica de *Hard Negative Patching* (HNP). Esta metodología, que consistió en reentrenar el modelo con ejemplos negativos difíciles —zonas del fondo visualmente ambiguas o similares a animales reales— permitió reducir de manera significativa los falsos positivos y reforzar la capacidad del modelo para discriminar entre regiones de interés y fondo. En particular, se observó una mejora en la precisión en clases que históricamente presentan alta tasa de confusión, como *kob*, *topi* y *buffalo*. El HNP también favoreció la estabilidad de las predicciones del modelo HerdNet+CBAM, y mejoró la claridad de los mapas de atención generados en la etapa de inferencia.

Estos resultados sugieren que el uso de estrategias de entrenamiento como HNP puede representar un factor diferencial en el rendimiento final del modelo, incluso cuando se

parte de arquitecturas con componentes más sofisticados como CBAM. Aunque su aplicación no fue suficiente para superar las métricas de trabajos anteriores, sí contribuyó a una mejor generalización del modelo en regiones complejas, y demostró ser una herramienta valiosa para escenarios con datos escasos o desbalanceados. En contextos donde el acceso a datos anotados es limitado, el refinamiento supervisado mediante ejemplos difíciles se perfila como una solución efectiva y complementaria a las mejoras arquitectónicas.

Por tanto, se propone que futuras investigaciones consideren la integración temprana de HNP u otras variantes como *Online Hard Example Mining* (OHEM) desde las primeras etapas del pipeline, así como su combinación con mecanismos de atención más adaptativos al dominio. La validación de este enfoque en escenarios operativos reales, con mayor diversidad de entornos y especies, será clave para consolidar su aplicabilidad en el monitoreo automatizado de fauna silvestre mediante visión computacional.

#### AGRADECIMIENTOS

Agradecemos al equipo de desarrollo y a los autores del artículo original por proporcionar el código base y dataset.

#### REFERENCES

- [1] A. Delplanque, S. Foucher, J. Théau, E. Bussière, C. Vermeulen, and P. Lejeune, "Multispecies detection and identification of African mammals in aerial imagery using convolutional neural networks," \*Remote Sensing in Ecology and Conservation\*, vol. 8, no. 2, pp. 166–179, 2022. [Online]. Available: <https://doi.org/10.1002/rse2.234>
- [2] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," \*arXiv preprint\*, arXiv:1709.01507, 2017. [Online]. Available: <https://arxiv.org/abs/1709.01507>
- [3] S. Woo, J. Park, J.-Y. Lee, and I. S. Kwon, "CBAM: Convolutional Block Attention Module," \*arXiv preprint\*, arXiv:1807.06521, 2018. [Online]. Available: <https://arxiv.org/abs/1807.06521>
- [4] A. Delplanque, S. Foucher, J. Théau, E. Bussière, C. Vermeulen, and P. Lejeune, "From crowd to herd counting: How to precisely detect and count African mammals using aerial imagery and deep learning?," \*ISPRS Journal of Photogrammetry and Remote Sensing\*, vol. 197, pp. 167–180, 2023. [Online]. Available: <https://doi.org/10.1016/j.isprsjprs.2023.01.025>
- [5] B. Kellenberger, M. Volpi, and D. Tuia, "Fast animal detection in UAV images using convolutional neural networks," in \*Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)\*, 2017, pp. 866–869. [Online]. Available: <https://doi.org/10.1109/IGARSS.2017.8127090>
- [6] A. Shrivastava, A. Gupta, and R. Girshick, "Training Region-based Object Detectors with Online Hard Example Mining," \*arXiv preprint\*, arXiv:1604.03540, 2016. [Online]. Available: <https://arxiv.org/abs/1604.03540>
- [7] Q. Luo, Z. Zhang, C. Yang and J. Lin, "An Improved Soft-CBAM-YoloV5 Algorithm for Fruits and Vegetables Detection and Counting," 2023 IEEE 6th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Haikou, China, 2023, pp. 187-192, doi: 10.1109/PRAI59366.2023.10332084.
- [8] S. Christin, É. Hervet, and N. Lecomte, "Applications for deep learning in ecology," *Methods in Ecology and Evolution*, vol. 10, no. 10, pp. 1632–1644, 2019. [Online]. Available: <https://doi.org/10.1111/2041-210X.13256>
- [9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *arXiv preprint arXiv:1506.01497*, 2015. [Online]. Available: <https://doi.org/10.48550/arXiv.1506.01497>
- [10] B. Kellenberger, M. Volpi and D. Tuia, "Fast animal detection in UAV images using convolutional neural networks," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 2017, pp. 866-869, doi: 10.1109/IGARSS.2017.8127090.
- [11] B. Kellenberger, D. Marcos, and D. Tuia, "Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning," *Remote Sensing of Environment*, vol. 216, pp. 139–153, Oct. 2018, doi: 10.1016/j.rse.2018.06.028.
- [12] A. A. Ayantunde, A. J. Duncan, M. T. van Wijk, and P. Thorne, "Review: Role of herbivores in sustainable agriculture in Sub-Saharan Africa," *Animal*, vol. 12, Suppl. 2, pp. s199–s209, 2018, doi: 10.1017/S175173111800174X.
- [13] A. A. Ayantunde, A. J. Duncan, M. T. van Wijk, and P. Thorne, "Review: Role of herbivores in sustainable agriculture in Sub-Saharan Africa," *Animal*, vol. 12, Suppl. 2, pp. s199–s209, Dec. 2018, doi: 10.1017/S175173111800174X.

#### VII. ANEXOS

- **Repositorio de HerdNet CBAM:** <https://github.com/victordpd19/HerdNet>
- **Repositorio de desarrollo:** [https://github.com/victordpd19/maia\\_proyecto\\_final](https://github.com/victordpd19/maia_proyecto_final)