



**UNIVERSIDAD
NACIONAL
DE MISIONES**



**FACULTAD
DE INGENIERÍA
UNaM**

Sistema de Monitoreo y Alerta Temprana basado en IA para Áreas Protegidas

Proyecto Final Integrador - Ingeniería en Computación

Fabrizio Contigiani - Gabriel Da Silva

Tutor: Dr. Ing. Sergio Moya

31 de diciembre de 2025

1. Contexto Regional
2. Sistemas Actuales
3. Motivacion y Objetivos
4. Marco Teorico
5. Arquitectura del Sistema
6. Metodologia
7. Pruebas y Resultados
8. Conclusiones

Contexto Regional


El Bosque Atlántico

- Originalmente: 1.3 millones de km² [1]
- Brasil (92 %), Paraguay (6 %), Argentina (2 %)
- Hoy: solo 12-17 % de su extensión original
- Uno de los **hotspots de biodiversidad** más amenazados del planeta [2]

[Mapa Bosque Atlántico
histórico vs. actual]

Remanente continuo mas extenso del Bosque Atlantico en el Cono Sur

- 1.1 millones de hectareas protegidas
- **Corredor Verde de Misiones**
- Mas del **50 % de la biodiversidad** de Argentina [3]
- En menos del 0.5 % del territorio nacional



[Mapa Selva Misionera
Corredor Verde]

Concentracion de especies:

- ~3,000 especies de plantas vasculares
- 554 especies de aves
- 120 especies de mamiferos
- 79 reptiles y 55 anfibios

Especies emblematicas:


- Yaguarete (*Panthera onca*)
- Tapir (*Tapirus terrestris*)
- Aguila harpia (*Harpia harpyja*)
- Yacutinga (*Aburria jacutinga*)

[Collage fauna emblematica]

El Yaguarete: Especie Bandera

- **Monumento Natural** provincial (1988) y nacional (2001)
- Censo 2024: ~84 individuos en el Corredor Verde [4]
- Menos de 250 adultos en toda Argentina
- **En peligro crítico** de extincion

Su presencia es **indicador clave** del estado de salud del ecosistema



[Foto yaguarete
camara trampa]

Servicios ecosistemicos:

- Regulacion del ciclo hidrológico
- Secuestro de carbono
- Protección contra la erosión del suelo
- Regulación climática regional

Marco legal: Ley 26.331 de Protección de Bosques Nativos [5]

Categoría I (Rojo): Muy alto valor - no se transforma

Categoría II (Amarillo): Mediano valor - uso sostenible

Categoría III (Verde): Bajo valor - puede transformarse

Amenazas a la Conservación



[Imagen amenazas a la conservacion]

- Deforestacion y fragmentacion del habitat
- Caza furtiva y trafico de fauna
- Intrusion en areas protegidas
- Recursos limitados para vigilancia

1990-2020: ~130,000 hectareas perdidas solo en el Corredor Verde [6]

- 77 % en parcelas <50 ha
- Ocupaciones para cultivos de subsistencia
- Tala ilegal de madera noble

2025: Reduccion del 18 % respecto al promedio historico [7]

[Mapa deforestacion
o grafico temporal]

Fragmentacion del Habitat

Consecuencias

- Poblaciones aisladas geneticamente
- Desplazamiento por areas no protegidas
- Conflictos con actividades humanas
- Atropellamientos en rutas

[Diagrama fragmentacion del habitat]

Dos dimensiones:

1. **Cultural/subsistencia:**
Residentes locales
2. **Economica/organizada:** Trafico de fauna

Zonas criticas:

- Frontera con Brasil
- Reserva de Biosfera Yaboti
- Parques provinciales Pinalito, Uruguay-i

Especies afectadas: Tapir, paca, corzuelas, tucanes, loros

[Imagen problematica
caza furtiva]

Actividades ilicitas frecuentes:

- Pesca ilegal en cursos de agua
- Desmonte encubierto para expansion de cultivos
- Campamentos de caza con infraestructura permanente
- Extraccion de recursos naturales

Problema critico

Sin un sistema de **alerta temprana**, las intrusiones se descubren **a posteriori** durante patrullajes de rutina, cuando el dano ya fue perpetrado.

Desafíos de la Vigilancia

780,000 hectareas distribuidas en **106+ areas protegidas**

Extension: Terreno accidentado, vegetacion densa, acceso limitado

Comunicaciones: Sin cobertura celular en zonas interiores

Latencia: Semanas/meses entre captura de evento y descubrimiento

Volumen: Miles de imagenes requieren clasificacion manual



[Foto vegetacion densa]

Sistemas Actuales


Camaras Trampa Tradicionales [8]

Componentes:

- Sensor PIR (movimiento)
- Camara digital
- Almacenamiento SD
- Iluminacion IR (nocturna)
- Baterias AA

Ventajas:

- Alta autonomia
- Bajo costo inicial
- Robustez probada



[Foto camara trampa tradicional]

Limitaciones de Camaras Tradicionales

Brechas criticas

Latencia: Imagenes almacenadas localmente por semanas/meses

Sin alertas: Informacion fluye solo hacia centros de analisis

Manual: 80 %+ de imagenes son vacias o irrelevantes [9]

Mantenimiento: Visitas periodicas para baterias y tarjetas

*Para cuando se detecta una intrusion,
los responsables ya estan lejos del area.*

Ejemplos: Tactacam REVEAL [10],
Spypoint Flex [11]

Características:

- Envio via redes 3G/4G/LTE
- App movil propietaria
- Disparo rapido ($<0.5s$)
- Vision nocturna IR

[Camara celular comercial]

Limitaciones de Soluciones Comerciales

Dependencia de infraestructura

- **Sin cobertura celular** en interior de reservas
- En selva densa, la señal es inexistente

Costos elevados

- Alto costo de adquisicion por unidad
- Suscripciones mensuales: \$5-15 USD por camara
- Servicios en la nube propietarios

Escalabilidad limitada

- Costo prohibitivo para despliegues masivos
- Especialmente en paises en vias de desarrollo

Barrero & Schmunck (UNaM, 2023) [12]: Microcamara de vigilancia para fauna - bases para soluciones de bajo costo

Whytock et al. (2023) [13]: Camaras trampa con IA + alertas satelitales Iridium - costos operativos muy altos

AiCatcher (Mallya, 2019) [14]: Raspberry Pi + LoRa + inferencia en borde - alto consumo energetico

Brecha identificada

Oportunidad para un sistema que combine **bajo costo, independencia de infraestructura celular, y alertas en tiempo real.**

Analisis Comparativo

Caracteristica	Tradicional	Celular	Satelital	LoRa
Alertas tiempo real	No	Si	Si	Si
Infraestructura	Ninguna	Operador	Satelite	Gateway
Procesamiento IA	Post-hoc	No/Limitado	Edge	Edge
Costo adquisicion	Bajo	Medio-Alto	Muy alto	Medio
Costo operativo	Bajo	Alto	Muy alto	Bajo
Autonomia	Muy alta	Media	Baja	Baja

Las alertas inmediatas estan condicionadas por elevados costos o dependencia de terceros.

Motivacion y Objetivos

Transformar el paradigma:

Monitoreo **pasivo** → Vigilancia **activa e inteligente**

Un sistema que:

- No solo capture imagenes, sino que las **transmita en tiempo real**
- Las **analice automaticamente** mediante IA
- Genere **alertas inmediatas** ante eventos de interes
- Sea **accesible y de bajo costo**
- Use **hardware economico y software de codigo abierto**

Convergencia de tres desarrollos:

1. Microcontroladores de bajo costo

ESP32: WiFi, Bluetooth, bajo consumo. Costo: <\$10 USD

2. Redes mesh autoorganizadas

ESP-MESH: Extension de cobertura sin infraestructura celular

3. IA para vision por computadora

SpeciesNet, MegaDetector: Modelos de codigo abierto para clasificacion automatica de fauna

Objetivo

Demostrar la **viabilidad tecnica** de un sistema de monitoreo de **bajo costo** para areas protegidas, basado en:

- Nodos de camara con conectividad **mesh**
- Clasificacion automatica de imagenes mediante **IA**

Reduccion del tiempo de procesamiento:

Dias/semanas → Minutos

Objetivos Especificos

1. Diseñar e implementar un **nodo de cámara autónomo** basado en ESP32
2. Desarrollar una **red mesh autoorganizada** con ESP-MESH
3. Implementar un **servidor de procesamiento** con IA
4. Integrar **SpeciesNet/MegaDetector** para clasificación
5. **Validar el funcionamiento** en condiciones reales

Dentro del alcance:

- Nodos ESP32-CAM + sensor PIR
- Red mesh funcional
- Servidor de recepcion y clasificacion
- Bot de Telegram para alertas
- Pruebas en laboratorio y campo

Fuera del alcance:

- Despliegue en selva densa
- Carcasas con grado IP
- Entrenamiento de modelos especificos
- Certificacion comercial

Limitaciones Conocidas

Operacion diurna: Lente con filtro IR, sin vision nocturna

Sensor PIR: Optimizado para humanos, puede no detectar fauna pequena

Resolucion: Limitada por tamano maximo de paquete mesh ($<8\text{KB}$)

Consumo: WiFi consume mas que camaras trampa comerciales optimizadas

Nota sobre “tiempo real”

No es tiempo real deterministico (milisegundos), sino **operativo:** de dias/semanas a **minutos**.

Marco Teorico

Funcionamiento de Camaras Trampa

Ciclo de operacion:

1. **Reposo:** Solo sensor PIR activo (bajo consumo)
2. **Deteccion:** Cambio en radiacion IR
3. **Activacion:** Despertar camara (trigger time: 0.1-2s)
4. **Captura:** Foto/video + metadatos
5. **Almacenamiento:** Compresion JPEG, grabacion en SD
6. **Retorno al reposo**



[Diagrama ciclo operacion]

Arquitectura de tres capas:



[Diagrama arquitectura IoT - 3 capas]

1. **Percepcion:** Sensores, camaras → ESP32-CAM + PIR
2. **Red:** Transmision de datos → ESP-MESH
3. **Aplicacion:** Procesamiento, alertas → Django + SpeciesNet

Protocolo	Alcance	Ancho banda	Consumo
BLE	~100m	Bajo	Muy bajo
ZigBee	~100m	Bajo	Bajo
LoRa	km	Muy bajo	Bajo
WiFi	~100m	Alto	Alto

LoRa: Excelente alcance, pero ancho de banda **insuficiente** para imagenes

WiFi Mesh: Permite transmision de imagenes + extension de cobertura

Características:

- **Autoorganizacion:** Selecccion automatica de padre optimo
- **Autocuracion:** Reconexion ante fallos
- Topologia de arbol
- Hasta 6+ niveles de profundidad
- Cientos de nodos

[Diagrama topologia mesh]

You Only Look Once

- Deteccion en una sola pasada
- Bounding boxes + probabilidades
- Balance velocidad/precision
- YOLOv5: PyTorch, multiples variantes [16]

[Ejemplo deteccion YOLO
con bounding boxes]

Microsoft AI for Earth

Modelo especializado para camaras trampa que detecta:

- **Animales** (sin distincion de especie)
- **Personas**
- **Vehiculos**

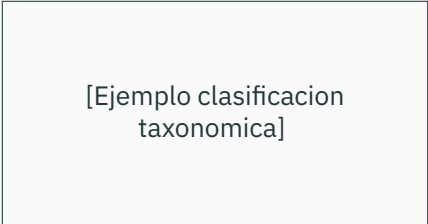
Ventajas:

- Altamente robusto y generalizable
- Adoptado por 60+ organizaciones de conservacion [18]
- Reduce hasta 90 % el tiempo de procesamiento

Pipeline de dos fases:

1. **Deteccion:** Identifica regiones de interes (animales, personas, vehiculos)
2. **Clasificacion taxonomica:** Para animales detectados
 - o Familia
 - o Genero
 - o Especie

Entrenado con millones de imagenes de camaras trampa a nivel global



[Ejemplo clasificacion
taxonomica]

Arquitectura del Sistema

[Diagrama arquitectura general del sistema]
Nodos camara → Red Mesh → Nodo Raiz → Servidor →
Telegram

1. Sensor PIR detecta movimiento → interrupcion
2. Nodo captura imagen JPEG (<8KB)
3. Transmision via red mesh hasta nodo raiz
4. Nodo raiz envia HTTP POST al servidor
5. Servidor procesa con SpeciesNet
6. Si hay deteccion: alerta via Telegram
7. Almacenamiento en base de datos

Componentes:

- ESP32-CAM (AI-Thinker)
- Camara OV2640 (2MP)
- Sensor PIR HC-SR501
- Buck converter LM2596
- Baterias 18650 (2S, 7.4V)

Resolucion: 640x480 (VGA)

[Foto nodo camara
ensamblado]

Desarrollado con ESP-IDF + ESP-MDF

Funcionalidades:

- Inicializacion de camara OV2640
- Interrupcion GPIO para sensor PIR
- Conexion a red ESP-MESH como nodo hijo
- Captura y compresion JPEG
- Almacenamiento en microSD (respaldo)
- Transmision via Mwifi
- Modo power-save para ahorro de energia

Diseno basado en modelo CC 4.0

Caracteristicas:

- Material: PLA
- Apertura para lente
- Domo para sensor PIR
- Espacio para buck converter
- Acceso a tarjeta SD
- Puntos de montaje

Disponible en Printables

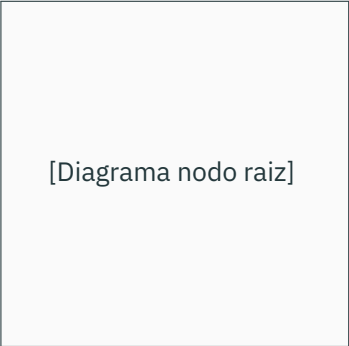
[Render/foto carcasa 3D]

Hardware: ESP32 DevKit V1 (sin camara)

Rol: Puerta de enlace Mesh ↔ Internet

Funciones:

- Raiz de la red ESP-MESH
- Recepcion de imagenes de nodos hijos
- Conexion WiFi a router
- Cliente HTTP para envio al servidor
- Sincronizacion de tiempo

A large, empty rectangular box with a thin black border, intended for a diagram of the root node.

[Diagrama nodo raiz]

Arquitectura containerizada con Docker Compose:

Django: Backend web, recepcion de imagenes

SpeciesNet: Servicio de inferencia (LitServe)

PostgreSQL: Base de datos

Telegram Bot: Notificaciones

A large rectangular box with a thin black border, containing the text "[Diagrama Docker Compose]".

[Diagrama Docker Compose]

Bot de Telegram

Cada notificacion incluye:

- Imagen original
- Imagen anotada (bounding boxes)
- Cantidad y confianza de detecciones
- Clasificaciones taxonomicas

Comandos:

- /start - Registro
- /last - Ultima imagen

[Screenshot notificacion
Telegram]

Metodologia

Iterativo e incremental

- Cada componente desarrollado, probado y refinado individualmente
- Identificación temprana de problemas
- Metodología ágil **Kanban**

Control de versiones: GitHub (4 repositorios)

Despliegue continuo: GitHub Actions

- Build automático de imágenes Docker
- Publicación en GitHub Container Registry

Etapas del Desarrollo

[Diagrama Gantt - etapas del desarrollo]

1. Investigacion y diseno
2. Desarrollo del firmware
3. Desarrollo del servidor
4. Diseno y fabricacion de hardware
5. Integracion
6. Pruebas y validacion

Herramientas Utilizadas

Desarrollo:

- VS Code
- ESP-IDF + ESP-MDF
- Django
- python-telegram-bot
- Docker / Docker Compose

Diseño 3D:

- Autodesk Fusion 360
- PrusaSlicer
- Impresora Creality Ender 3

Desafios Enfrentados

Desafio 1: Limite de tamanho de pacote

ESP-MDF limita pacotes a 8KB → ajustar compressão JPEG

Desafio 2: Consumo energético

WiFi consome significativamente → modo power-save de Mwifi

Desafio 3: Estabilidade de rede mesh

Reconexões frequentes → tuning de parâmetros de conexão

Desafio 4: Tempo de inferência

SpeciesNet lento em CPU → suporte para GPU
(nvidia-container-toolkit)

Pruebas y Resultados

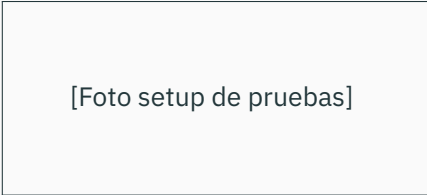
Configuracion de Pruebas

Hardware:

- 3 nodos de camara
- 1 nodo raiz
- Baterias 18650 (2S)
- Router WiFi domestico

Software:

- ESP-IDF v5.x
- Docker containers
- SpeciesNet + LitServe
- PostgreSQL
- Tunel VPN (Pangolin)



[Foto setup de pruebas]

Validaciones realizadas:

- Captura de imagen: inicializacion, calidad, tamaño
- Sensor PIR: distancia, tiempo de respuesta, falsas activaciones
- Transmision mesh: conexion, latencia, perdida de paquetes
- Clasificacion: deteccion de animales/personas/vehiculos
- Alertas: recepcion en Telegram, contenido correcto

Resultados - Captura de Imagen



[Ejemplo imagen capturada 640x480]

- Resolución: 640x480 (VGA)
- Formato: JPEG comprimido
- Tamaño típico: <8KB

Resultados - Red Mesh

Metrica	Valor
Latencia promedio	[X] s
Distancia max (linea de vista)	[X] m
Distancia max (con obstaculos)	[X] m
Tasa perdida de paquetes	<5 %
Tiempo de reconexion	[X] s

Categoria	Precision
Animales	[X] %
Personas	[X] %
Vehiculos	[X] %

Tiempo de inferencia:

- CPU: 1-5 s/imagen
- GPU: 0.5-1 s/imagen

[Ejemplo deteccion exitosa]

Resultados - Consumo Energetico

Estado	Consumo (mA)
Modo power-save	[X]
Captura de imagen	[X]
Transmision WiFi	[X]
Promedio ponderado	[X]


Autonomia estimada: [X] horas con baterias 18650 (2S, [X]mAh)

Condiciones:

- Duracion: [X] horas
- 3 nodos desplegados
- Distancia entre nodos: [X]m
- Condiciones climaticas favorables

Resultados:

- Imagenes capturadas: [X]
- Detecciones exitosas: [X]
- Falsas activaciones: [X]



[Foto nodo desplegado en campo]

Conclusiones

Logros Alcanzados

- ✓ Sistema completo de monitoreo con **hardware de bajo costo**
- ✓ Reduccion de tiempo de procesamiento: **dias** → **minutos**
- ✓ Red mesh funcional con **ESP-MESH**
- ✓ Integracion exitosa de **SpeciesNet/MegaDetector**
- ✓ Sistema de alertas en tiempo real via **Telegram**
- ✓ Carcasa imprimible en 3D
- ✓ **Codigo abierto** disponible en GitHub

Aportes del Trabajo

Arquitectura integrada

Captura distribuida + transmision mesh + procesamiento IA centralizado

Prototipo de bajo costo

Componentes comerciales economicos + disenos 3D listos para fabricacion

Integracion de SpeciesNet

Pipeline de clasificacion taxonomica en tiempo operativo

Codigo abierto

4 repositorios publicos en GitHub

Comparacion con Objetivos

Metrica	Objetivo	Resultado
Latencia de respuesta	$< X$ min	$[X]$ min
Precision de deteccion	$> X$ %	$[X]$ %
Autonomia	$> X$ horas	$[X]$ horas
Cobertura	$> X$ m	$[X]$ m

Vision nocturna: Camaras con capacidad IR

Conectividad largo alcance: LoRa, 4G/LTE, satelital

Optimizacion energetica: Paneles solares, nodo repetidor dedicado

Procesamiento en borde: TinyML en nodos de camara

Modelos especificos: Entrenamiento para fauna regional

Interfaz web: Mapas, gestion de despliegues, reportes

1. **Ubicaciones:** Considerar distancia entre nodos y obstaculos
2. **Proteccion:** Usar PETG/ASA para despliegues permanentes
3. **Baterias:** Dimensionar segun frecuencia de activaciones
4. **Sensor PIR:** Evaluar alternativas para fauna pequena
5. **Monitoreo:** Implementar seguimiento de estado de nodos

mesh-node: `github.com/fabcontigiani/
mesh-node-capstone-project`

root-node: `github.com/fabcontigiani/
root-node-capstone-project`

server: `github.com/fabcontigiani/
server-capstone-project`

wildlife-detection: `github.com/fabcontigiani/
wildlife-detection-capstone-project`

Modelo 3D carcasa: Disponible en Printables

Agradecimientos

- A nuestras familias por el apoyo incondicional
- Al Dr. Ing. Sergio Moya por su guía y dedicación como tutor
- A la Facultad de Ingeniería de la UNaM
- A los docentes que contribuyeron a nuestra formación
- A nuestros compañeros de carrera

¡Gracias a todos!

¿Preguntas?

- [1] M. C. Ribeiro, J. P. Metzger, A. C. Martensen, F. J. Ponzoni y M. M. Hirota, «The Brazilian Atlantic Forest: How much is left, and how is the remaining forest distributed? Implications for conservation,» *Biological Conservation*, vol. 142, n.º 6, págs. 1141-1153, 2009. DOI: 10.1016/j.biocon.2009.02.021.
- [2] World Wildlife Fund, «Atlantic Forest: The most threatened tropical forest,» visitado 27 de dic. de 2024. dirección: <https://www.worldwildlife.org/places/atlantic-forest>.
- [3] M. S. Di Bitetti, G. Placci y L. A. Dietz, «A Biodiversity Vision for the Upper Paraná Atlantic Forest Ecoregion: Designing a Biodiversity Conservation Landscape and Setting Priorities for Conservation Action,» *World Wildlife Fund*, 2003.

Referencias ii

- [4] Fundación Vida Silvestre Argentina, «Censo de Yagareté 2024: Resultados del monitoreo binacional en el Corredor Verde,» visitado 27 de dic. de 2024. dirección: <https://www.vidasilvestre.org.ar/>.
- [5] Congreso de la Nación Argentina, *Ley 26.331 de Presupuestos Mínimos de Protección Ambiental de los Bosques Nativos*, Boletín Oficial de la República Argentina, Sancionada el 28 de noviembre de 2007, 2007.
- [6] Facultad de Agronomía, Universidad de Buenos Aires, «Estudio sobre pérdida de bosque nativo en el Corredor Verde de Misiones (1990-2020),» visitado 27 de dic. de 2024. dirección: <https://www.agro.uba.ar/>.
- [7] Ministerio de Ecología y Recursos Naturales Renovables de Misiones, «Informe de Deforestación 2025: Misiones alcanzó el nivel más bajo de pérdida de bosque nativo,» visitado 27 de dic. de 2024. dirección: <https://ecologia.misiones.gob.ar/>.

- [8] R. Steenweg et al., «Scaling-up camera traps: monitoring the planet's biodiversity with networks of remote sensors,» *Frontiers in Ecology and the Environment*, vol. 15, n.º 1, págs. 26-34, 2017. DOI: 10.1002/fee.1448.
- [9] M. A. Tabak et al., «Machine learning to classify animal species in camera trap images: Applications in ecology,» *Methods in Ecology and Evolution*, vol. 10, n.º 4, págs. 585-590, 2019. DOI: 10.1111/2041-210X.13120.
- [10] Tactacam, «Tactacam REVEAL Cellular Cameras,» visitado 27 de dic. de 2024. dirección: <https://www.revealcellcam.com/>.
- [11] Spypoint, «Spypoint Cellular Trail Cameras,» visitado 27 de dic. de 2024. dirección: <https://www.spypoint.com/>.

- [12] G. Barrero y A. Schmunck, «Desarrollo de una microcámara de vigilancia para la protección de la fauna salvaje,» Tesis de Grado, Universidad Nacional de Misiones, 2023. dirección: <https://drive.google.com/file/d/1jJDM-Mc8kHJw3Rcgl0tf4Mq3MxNyqXW5>.
- [13] R. C. Whytock et al., «Real-time alerts from AI-enabled camera traps using the Iridium satellite network: A case-study in Gabon, Central Africa,» *Methods in Ecology and Evolution*, vol. 14, n.º 3, págs. 867-874, 2023, ISSN: 2041-210X. DOI: 10.1111/2041-210X.14036. dirección: <http://dx.doi.org/10.1111/2041-210X.14036>.
- [14] D. Mallya, «AiCatcher: Extending machine intelligence into the wild,» visitado 27 de dic. de 2024. dirección: <https://deepakmallya.com/aicatcher>.

- [15] Espressif Systems, «ESP-MESH Programming Guide,» visitado 15 de ene. de 2024. dirección:
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-guides/esp-wifi-mesh.html>.
- [16] G. Jocher, «YOLOv5 by Ultralytics,» Repositorio oficial de YOLOv5, visitado 27 de dic. de 2024. dirección:
<https://github.com/ultralytics/yolov5>.
- [17] J. Redmon, S. Divvala, R. Girshick y A. Farhadi, «You Only Look Once: Unified, Real-Time Object Detection,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, págs. 779-788, 2016.

- [18] J. Vélez et al., «An evaluation of platforms for processing camera-trap data using artificial intelligence,» *Methods in Ecology and Evolution*, vol. 14, n.º 2, págs. 459-477, 2022, ISSN: 2041-210X. DOI: 10.1111/2041-210X.14044.
dirección: <http://dx.doi.org/10.1111/2041-210X.14044>.
- [19] S. Beery, D. Morris y S. Yang, «Efficient Pipeline for Camera Trap Image Review,» *arXiv preprint arXiv:1907.06772*, 2019.
dirección: <https://arxiv.org/abs/1907.06772>.
- [20] T. Gadot et al., «To crop or not to crop: Comparing whole-image and cropped classification on a large dataset of camera trap images,» *IET Computer Vision*, 2024.

[Esquemático KiCad del nodo]

Planos de la Carcasa

[Planos 2D de la carcasa]

Inversión inicial: \$20,530 USD

Modelo de negocio:

- Cobro por instalación
- Suscripción mensual (\$50 USD)

Mercado objetivo:

- 116 reservas naturales
- 10,800+ EAPs con bosques

Indicadores:

- VAN: \$7,517.55 USD
- TIR: 25 %
- TREMA: 15 %
- Recupero: Año 4

Conclusion:

Proyecto **economicamente viable**

[Graficos de análisis de sensibilidad]

- Punto de equilibrio: 24 suscripciones/año
- Precio mínimo viable: \$40 USD/mes