

Sistema de Monitoreo y Alerta Temprana basado en Inteligencia Artificial para Áreas Protegidas

Autores:

Fabrizio Martin Contigiani
Gabriel Orlando Da Silva Schmies

Tutor:

Dr. Ing. Sergio Eduardo Moya

27 de diciembre de 2025

Índice general

Índice de figuras	2
Índice de cuadros	3
Glosario	4
Siglas	5
1. Introducción	6
1.1. Contexto y motivación	6
1.2. Planteamiento del problema	6
1.3. Justificación	6
1.4. Objetivos	6
1.4.1. Objetivo general	6
1.4.2. Objetivos específicos	6
1.5. Alcance y limitaciones	6
1.6. Estructura del documento	6
2. Marco Teórico	7
2.1. Áreas protegidas y conservación de fauna silvestre	8
2.2. Sistemas de vigilancia y monitoreo	8
2.2.1. Cámaras trampa tradicionales	8
2.2.2. Sistemas de detección de intrusos	8
2.3. Internet de las Cosas (IoT)	8
2.3.1. Arquitecturas IoT	8
2.3.2. Protocolos de comunicación inalámbrica	8
2.4. Redes Mesh	8
2.4.1. Topologías de red	8
2.4.2. ESP-MESH y Mwifi	8
2.5. Inteligencia Artificial aplicada a visión por computadora	8
2.5.1. Redes neuronales convolucionales (CNN)	8
2.5.2. Detección de objetos con YOLO	8
2.5.3. SpeciesNet de Google	8
2.6. Tecnologías de desarrollo	8
2.6.1. Microcontroladores ESP32	8
2.6.2. ESP-IDF y ESP-MDF	8
2.6.3. Contenedorización con Docker	8

2.6.4. Framework Django	8
3. Antecedentes	9
3.1. Trabajos relacionados	9
3.2. Soluciones comerciales existentes	9
3.3. Estado del arte	9
3.4. Análisis comparativo	9
4. Metodología	10
4.1. Enfoque metodológico	10
4.2. Etapas del desarrollo	10
4.3. Herramientas y tecnologías utilizadas	10
4.4. Métricas de evaluación	10
5. Diseño del Sistema	11
5.1. Arquitectura general	12
5.2. Diseño del hardware	12
5.2.1. Selección de componentes	12
5.2.2. Nodo de captura con cámara	12
5.2.3. Nodo raíz	12
5.2.4. Alimentación y consumo energético	12
5.3. Diseño de la red mesh	12
5.3.1. Topología de la red	12
5.3.2. Protocolo de comunicación	12
5.3.3. Formato de datos	12
5.4. Diseño del servicio de detección	12
5.4.1. Servidor de inferencia con SpeciesNet	12
5.4.2. Detección de animales, humanos y vehículos	12
5.4.3. Anotación de imágenes con bounding boxes	12
5.5. Diseño del servidor de aplicación	12
5.5.1. Arquitectura de servicios	12
5.5.2. Gestión de imágenes	12
5.5.3. Interfaz web	12
5.5.4. Bot de Telegram y sistema de alertas	12
6. Implementación	13
6.1. Nodo mesh (mesh-node)	14
6.1.1. Firmware del nodo de captura	14
6.1.2. Captura de imágenes	14
6.1.3. Compresión y transmisión	14
6.2. Nodo raíz (root-node)	14
6.2.1. Firmware del nodo raíz	14
6.2.2. Conexión con servidor TCP	14
6.2.3. Gestión de la red mesh	14
6.3. Servicio de detección (wildlife-detection)	14
6.3.1. Contenedor Docker con SpeciesNet	14
6.3.2. API de inferencia con LitServe	14

6.3.3. Procesamiento de imágenes	14
6.4. Servidor de aplicación (server)	14
6.4.1. Aplicación Django	14
6.4.2. Integración con SpeciesNet	14
6.4.3. Bot de Telegram y sistema de alertas	14
6.4.4. Base de datos PostgreSQL	14
6.4.5. Despliegue con Docker Compose	14
7. Pruebas y Resultados	15
7.1. Ambiente de pruebas	15
7.2. Pruebas de conectividad y red mesh	15
7.2.1. Alcance de la red	15
7.2.2. Latencia de transmisión	15
7.2.3. Estabilidad de la conexión	15
7.3. Pruebas de detección	15
7.3.1. Detección de fauna silvestre	15
7.3.2. Detección de humanos	15
7.3.3. Detección de vehículos	15
7.4. Evaluación del modelo de IA	15
7.4.1. Precisión y recall	15
7.4.2. Tiempo de inferencia	15
7.5. Pruebas de consumo energético	15
7.6. Pruebas del sistema de alertas	15
7.6.1. Tiempo de respuesta	15
7.7. Análisis de resultados	15
8. Conclusiones	16
8.1. Conclusiones generales	16
8.2. Aportes del trabajo	16
8.3. Trabajos futuros	16
8.4. Recomendaciones	16
Bibliografía	17
A. Esquemáticos del hardware	18
B. Código fuente relevante	19
B.1. Firmware del nodo mesh	19
B.2. Firmware del nodo raíz	19
B.3. Servidor de detección	19
B.4. Aplicación Django	19
C. Manual de instalación y configuración	20
C.1. Configuración del firmware	20
C.2. Despliegue del servidor	20
C.3. Configuración del bot de Telegram	20
D. Manual de usuario	21

E. Especificaciones técnicas	22
F. Análisis de viabilidad económica	23

Resumen

Esta primera versión del borrador se ha centrado primordialmente en la definición de la estructura general y la articulación del contenido central del documento. En consecuencia, se priorizaron estos aspectos, dejando la implementación de detalles de formato, tales como la carátula, el encabezado y el pie de página, para una próxima iteración o entrega. Asimismo, las secciones que aún requieren de ser escritas han sido completadas temporalmente con texto de relleno para poder visualizar correctamente la estructura del documento.

Palabras Clave - Cámaras Trampa, Internet de las Cosas, Inteligencia Artificial, Monitoreo de Fauna, Detección de Intrusos, Vigilancia, Wi-Fi Mesh

Índice general

Índice de figuras

Índice de cuadros

Glosario

miniz Librería escrita en C de implementación ligera para compresión y descompresión de datos basada en el algoritmo zlib. 6

Siglas

IA Inteligencia Artificial. 6

Capítulo 1

Introducción

Inteligencia Artificial (IA) miniz [1]

- 1.1. Contexto y motivación
- 1.2. Planteamiento del problema
- 1.3. Justificación
- 1.4. Objetivos
 - 1.4.1. Objetivo general
 - 1.4.2. Objetivos específicos
- 1.5. Alcance y limitaciones
- 1.6. Estructura del documento

Capítulo 2

Marco Teórico

2.1. Áreas protegidas y conservación de fauna silvestre

2.2. Sistemas de vigilancia y monitoreo

2.2.1. Cámaras trampa tradicionales

2.2.2. Sistemas de detección de intrusos

2.3. Internet de las Cosas (IoT)

2.3.1. Arquitecturas IoT

2.3.2. Protocolos de comunicación inalámbrica

2.4. Redes Mesh

2.4.1. Topologías de red

2.4.2. ESP-MESH y Mwifi

2.5. Inteligencia Artificial aplicada a visión por computadora

2.5.1. Redes neuronales convolucionales (CNN)

2.5.2. Detección de objetos con YOLO

2.5.3. SpeciesNet de Google

2.6. Tecnologías de desarrollo

2.6.1. Microcontroladores ESP32

2.6.2. ESP-IDF y ESP-MDF

2.6.3. Contenedorización con Docker

2.6.4. Framework Django

Capítulo 3

Antecedentes

- 3.1. Trabajos relacionados**
- 3.2. Soluciones comerciales existentes**
- 3.3. Estado del arte**
- 3.4. Análisis comparativo**

Capítulo 4

Metodología

- 4.1. Enfoque metodológico
- 4.2. Etapas del desarrollo
- 4.3. Herramientas y tecnologías utilizadas
- 4.4. Métricas de evaluación

Capítulo 5

Diseño del Sistema

5.1. Arquitectura general

5.2. Diseño del hardware

5.2.1. Selección de componentes

5.2.2. Nodo de captura con cámara

5.2.3. Nodo raíz

5.2.4. Alimentación y consumo energético

5.3. Diseño de la red mesh

5.3.1. Topología de la red

5.3.2. Protocolo de comunicación

5.3.3. Formato de datos

5.4. Diseño del servicio de detección

5.4.1. Servidor de inferencia con SpeciesNet

5.4.2. Detección de animales, humanos y vehículos

5.4.3. Anotación de imágenes con bounding boxes

5.5. Diseño del servidor de aplicación

5.5.1. Arquitectura de servicios

5.5.2. Gestión de imágenes

5.5.3. Interfaz web

5.5.4. Bot de Telegram y sistema de alertas

Capítulo 6

Implementación

6.1. Nodo mesh (mesh-node)

- 6.1.1. Firmware del nodo de captura
- 6.1.2. Captura de imágenes
- 6.1.3. Compresión y transmisión

6.2. Nodo raíz (root-node)

- 6.2.1. Firmware del nodo raíz
- 6.2.2. Conexión con servidor TCP
- 6.2.3. Gestión de la red mesh

6.3. Servicio de detección (wildlife-detection)

- 6.3.1. Contenedor Docker con SpeciesNet
- 6.3.2. API de inferencia con LitServe
- 6.3.3. Procesamiento de imágenes

6.4. Servidor de aplicación (server)

- 6.4.1. Aplicación Django
- 6.4.2. Integración con SpeciesNet
- 6.4.3. Bot de Telegram y sistema de alertas
- 6.4.4. Base de datos PostgreSQL
- 6.4.5. Despliegue con Docker Compose

Capítulo 7

Pruebas y Resultados

7.1. Ambiente de pruebas

7.2. Pruebas de conectividad y red mesh

7.2.1. Alcance de la red

7.2.2. Latencia de transmisión

7.2.3. Estabilidad de la conexión

7.3. Pruebas de detección

7.3.1. Detección de fauna silvestre

7.3.2. Detección de humanos

7.3.3. Detección de vehículos

7.4. Evaluación del modelo de IA

7.4.1. Precisión y recall

7.4.2. Tiempo de inferencia

7.5. Pruebas de consumo energético

7.6. Pruebas del sistema de alertas

7.6.1. Tiempo de respuesta

7.7. Análisis de resultados

Capítulo 8

Conclusiones

- 8.1. Conclusiones generales**
- 8.2. Aportes del trabajo**
- 8.3. Trabajos futuros**
- 8.4. Recomendaciones**

Bibliografía

- [1] 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.

Apéndice A

Esquemáticos del hardware

Apéndice B

Código fuente relevante

- B.1. Firmware del nodo mesh
- B.2. Firmware del nodo raíz
- B.3. Servidor de detección
- B.4. Aplicación Django

Apéndice C

Manual de instalación y configuración

C.1. Configuración del firmware

C.2. Despliegue del servidor

C.3. Configuración del bot de Telegram

Apéndice D

Manual de usuario

Apéndice E

Especificaciones técnicas

Apéndice F

Análisis de viabilidad económica