



Escuela
Politécnica
Superior

Reconocimiento de aves analizando su vuelo



Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:

Sebastián Matheu Martínez

Tutores:

José García Rodríguez

Esther Sebastián González

Julio 2022

Reconocimiento de aves analizando su vuelo

Autor

Sebastián Matheu Martínez

Tutores

José García Rodríguez

Tecnología Informática y Computación

Esther Sebastián González

Ecología



Grado en Ingeniería Informática



Escuela
Politécnica
Superior



Universitat d'Alacant
Universidad de Alicante

ALICANTE, Julio 2022

Preámbulo

Este trabajo tiene como propósito ayudar al estudio de la población de aves en los humedales de Alicante utilizando técnicas de Deep Learning (DL) que permitan detectar e identificar especímenes capturados por una cámara panorámica. Se utilizará una transmisión de vídeo del Parque natural de las Lagunas de La Mata y Torrevieja para obtener datos, realizar entrenamientos pertinentes y comprobaciones de funcionamiento.

Agradecimientos

Me gustaría agradecer en primer lugar a mis tutores José García y Esther Sebastián por la paciencia, el apoyo y la ayuda prestada durante el desarrollo de este trabajo.

En segundo lugar al personal del parque natural que nos ha dado el permiso y acceso a las cámaras usadas.

Y por último a mi familia, que me ha dado la motivación y brindado el privilegio de poder formarme como profesional.

Índice general

1	Introducción	1
2	Objetivos	3
3	Marco Teórico	5
3.1	DL	5
3.1.1	Redes neuronales convolucionales	6
3.2	Aplicaciones reales	7
3.2.1	BirdNET	7
3.2.2	Internet of Birds	7
3.2.3	Merlin	7
4	Metodología	9
4.0.1	Software	9
4.0.2	OpenCV	9
4.0.2.1	PyTorch	9
4.0.2.2	YOLO	9
4.0.2.3	Protocolo de transmisión en tiempo real	10
4.0.3	Hardware	10
4.0.3.1	Cámara	10
5	Desarrollo y resultados	13
5.1	Detector de aves	13
5.1.1	Limitaciones y problemas	14
5.2	Generador de trayectorias	15
5.3	Dataset	18
6	Conclusiones	19
6.1	Ampliaciones futuras	19
	Bibliografía	21

Índice de figuras

3.1	Esquema de la arquitectura de una Redes neuronales convolucionales (CNN) básica.	6
3.2	Representación del flujo de trabajo de BirdNET.	7
4.1	Arquitectura de YOLOv5s.	11
5.1	Flujo de trabajo del proyecto.	13
5.2	Pájaro detectado en tabla 5.1.	14
5.3	Pájaro detectado correctamente.	15
5.4	Lluvia durante la noche.	16
5.5	Pájaro detectado como otra clase.	16
5.6	Pájaro no detectado.	17
5.7	Trayectoria detectada de un ave.	17
5.8	Trayectoria detectada de dos aves.	17

1 Introducción

Los humedales son uno de los grandes patrimonios ecológicos de España donde habita una gran diversidad de fauna y flora. Desgraciadamente, estas zonas son de las más amenazadas en el mundo al haber perdido entre un 60-70% de su superficie en el último siglo, lo que hace que en estas zonas delicadas se requiera un plan de conservación a largo plazo para evitar su deterioro y la perdida de biodiversidad a causa del turismo, la contaminación, sobreexplotación y su uso no sostenible.

En los últimos años, técnicas de Machine Learning (ML) han comenzado a ser usadas para apoyar investigaciones científicas de diversos campos gracias a su flexibilidad y el resultado positivo que se puede obtener de ellas Takeki, Trinh, Yoshihashi, Kawakami, y cols. (2016). Principalmente, se utilizan métodos de aprendizaje profundo que pese a ser más costosos de preparar al requerir un gran volumen de datos apropiadamente etiquetados y procesados, producen predicciones con un alto grado de fiabilidad.

En el caso de la ecología animal, la sobreabundancia de los datos que se pueden obtener gracias a cámaras de vídeo y fototrampeo o grabadoras automáticas de sonido pueden ser utilizadas para crear modelos que ayuden al estudio de patrones de comportamiento, abundancia o recuento de la especie o la distribución de la población en un área dada.

Para ayudar a la conservación y uso correcto de estas áreas naturales se propone el uso de tecnologías de visión por computador y aprendizaje profundo que ayude a la detección y clasificación de la población aviar de Alicante, por ejemplo, el flamenco (*Phoenicopterus ruber*), la cigüeña (*Himantopus himantopus*) o el chorlitejo patinegro (*Charadrius alexandrinus*), entre otros.

2 Objetivos

El objetivo principal de este trabajo es realizar un análisis del estado del arte en técnicas de seguimiento de aves en vídeos mediante técnicas de aprendizaje profundo para crear un marco de investigación y desarrollar un entorno que permita la adquisición del flujo de video, detección de aves y el análisis de las trayectorias descritas en el campo de visión de la cámara para el desarrollo de un dataset de trayectorias de vuelo que caracterizan a las aves que habitan en los humedales de Alicante.

Para ello se desplegará un sistema de captura de vídeo para la monitorización permanente del espacio natural y la adquisición y distribución del flujo de video para su procesamiento.

Además se diseñará de una arquitectura de procesamiento del flujo de vídeo para identificar especies y seguir tanto la trayectoria a lo largo del tiempo como su posible forma de aleteo, dependiendo de la distancia y nitidez de la imagen.

3 Marco Teórico

El uso más común de DL en la investigación ecológica se centra en tareas de clasificación supervisada, prediciendo la clase de varias entidades biológicas a partir de datos identificativos como imágenes o sonido Takeki, Trinh, Yoshihashi, Rei, y cols. (2016), Huang y Basanta (2019), Kahl y cols. (2021b).

Los avances en estas tecnologías, especialmente el desarrollo y mejora de rendimiento de CNN Brodrick y cols. (2019), es lo que ha popularizado su uso. Pese a que una de sus grandes desventajas sea el gran volumen de datos necesario para obtener un funcionamiento razonable, el reconocimiento de imágenes puede combinarse con otras tecnologías (por ejemplo, sensores de movimiento) para desarrollar procedimientos automatizados de procesamiento de datos para recoger y etiquetar datos que permitan crear un dataset. Además, la clasificación no tiene por qué limitarse a considerar una sola fuente de datos. Se ha demostrado que la combinación con información contextual sobre la ubicación, la fecha, el tiempo o el hábitat mejora la precisión de la predicción Cooke y cols. (2004), Ditria y cols. (2020).

En la actualidad, es común el uso de sistemas de cámaras de fototrampeo que monitorizan la fauna de áreas de interés, obteniendo imágenes de los especímenes, reconociéndolos y manteniendo estadísticas de la población, y también puede permitir la identificación, clasificación y posterior reidentificación de animales concretos.

También su uso se está extendiendo a iniciativas científicas comunitarias en forma de aplicaciones móviles que contribuyen a la identificación de especies y sobretodo a la recolección de nuevos datos con los que perfilar el rendimiento de estos sistemas. Un ejemplo destacable es BirdNET Kahl y cols. (2021a) que se comentará más adelante.

El reconocimiento de aves actualmente tiene dos vías de implementación principales: su reconocimiento mediante una imagen clara del espécimen o una grabación de su canto transformada a un espectrograma, ambas mediante el uso extensivo de CNN. Sobre el estudio de patrones de vuelo en solitario o en bandada encontramos información muy escasa, principalmente descripciones en páginas educativas sobre el avistamiento de pájaros Akçay y cols. (2020).

3.1 DL

Entendemos el aprendizaje profundo Goodfellow y cols. (2016) como un conjunto de técnicas y algoritmos que pueden generar automáticamente modelos predictivos a partir de datos aprendiendo las características necesarias para clasificarlos. El aprendizaje se puede realizar sin supervisión dejando que el sistema utilice datos no etiquetados y encuentre patrones y características por su cuenta en cada capa de su estructura, sin un resultado esperado y con el objetivo de , o supervisado junto con datos apropiadamente etiquetados con el resultado previsto para obtener un modelo generalizado que pueda aproximar una clasificación correcta.

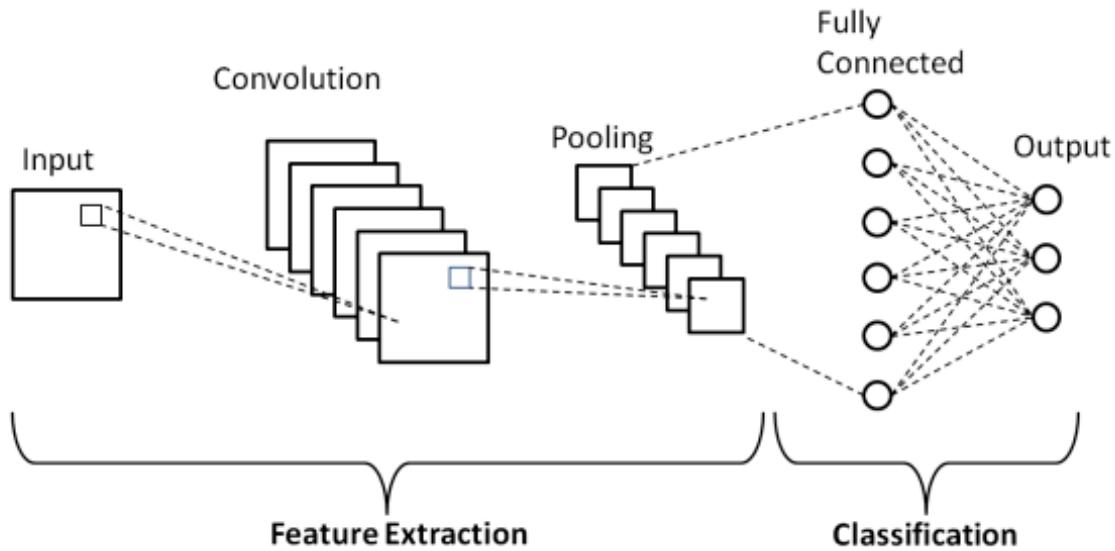


Figura 3.1: Esquema de la arquitectura de una CNN básica.

3.1.1 Redes neuronales convolucionales

En general, las CNN son un tipo de red neuronal artificial especializada en detectar patrones lo que las hace especialmente competentes para el análisis de imágenes. Estas se diferencian de otras redes tradicionales en que dentro de su arquitectura (figura 3.1) encontramos una o más capas de convolución dónde se especifica la cantidad de filtros que deben tener para descubrir características concretas de los datos procesados. Estos filtros son unas matrices de pequeño tamaño con las que se realiza la propia operación matemática de la convolución (que representa la forma en la que una función modifica o desplaza a otra) y dependiendo de su profundidad dentro de la red son más o menos complejos, detectando bordes o figuras geométricas en las capas menos profundas hasta caras o animales concretos en las más profundas.

Por ejemplo, podríamos definir un filtro de 5x5 para una capa de una red que trabaja con imágenes. Para aplicarlo, se realizaría el producto escalar de nuestro filtro con cada bloque de píxeles del mismo tamaño de la imagen que se pasa como datos de entrada. el resultado obtenido será utilizado por las siguientes capas como nueva entrada.

Las CNN no solo emplea capas convolucionales para su funcionamiento, si no que necesita de otras para desempeñar su trabajo correctamente. Una de las más comunes son las capas de pooling con las que se reduce el tamaño de las entradas haciendo que posteriores convoluciones analicen más cantidad de datos por operación t se reduzca la cantidad de parámetros dentro de la red mejorando su rendimiento computacional.

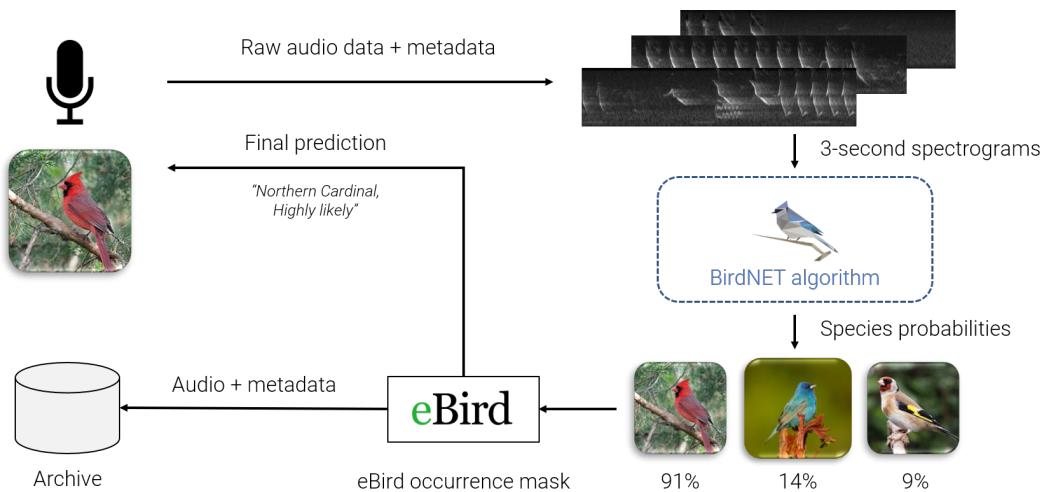


Figura 3.2: Representación del flujo de trabajo de BirdNET.

3.2 Aplicaciones reales

3.2.1 BirdNET

Esta es una aplicación de móvil Kahl y cols. (2021b) que implementa una CNN homónima, fruto de la investigación del Laboratorio de ornitología de la Universidad de Cornell y la Universidad Técnica de Chemnitz. Funciona realizando grabaciones en tiempo real del entorno, transformando el sonido en un espectrograma y enviándolo a un servidor para ser analizada por la red. Se utilizaron datos sacados des eBird para seleccionar las especies y grabaciones de Xeno-canto y la Biblioteca Macaulay para entrenar la red con un total de 226078 archivos para 984 especies de aves distintas. Además, se agregaron sonidos comunes a escuchar junto a los pájaros a forma de evitar falsos positivos: sonidos de otros animales, habla humana, silbidos, sonidos ambientales naturales como viento, lluvia, truenos, y urbanos como coches, aviones, sirenas, y ruido electrónico propio de una grabación.

3.2.2 Internet of Birds

Este proyecto orientado al hobby del avistamiento de pájaros es una aplicación móvil dedicada al reconocimiento de aves endémicas de Taiwán. El sistema funciona con una CNN alojada en un servidor, entrenada con un dataset propio de 3563 imágenes de veintisiete especies de la zona, añadiendo saltos de conectividad a la red para mejorar sus resultados. Su flujo de trabajo consiste en recibir una foto realizada por el usuario e intentar identificar el ave a partir de ella.

3.2.3 Merlin

Este es otro proyecto de la Universidad de Cornell orientado al avistamiento de aves que combina la funcionalidad de las aplicaciones anteriores y expande sus características permitiendo realizar una identificación manual a través de preguntas de las peculiaridades del pájaro y tener un listado de las aves reconocidas.

4 Metodología

En este capítulo se describen las herramientas software y hardware empleadas en el desarrollo del proyecto.

4.0.1 Software

En esta sección se describen las herramientas software relacionadas con visión artificial y adquisición de video.

4.0.2 OpenCV

OpenCV es una librería de código abierto de visión por computador y ML ampliamente usada en empresas, grupos de investigación y organismos gubernamentales desarrollada para hacer de base común en aplicaciones y productos comerciales que requieran uso de esta tecnología.

Esta librería contiene más de dos mil quinientos algoritmos optimizados, tanto clásicos como de última generación que pueden usarse para detectar y reconocer caras, identificar objetos, producir nubes de puntos tridimensionales, reconocimiento de paisajes, sustracción del fondo de una imagen, entre otros. Dentro de sus aplicaciones en el mundo real, se usa activamente para unir imágenes entre ellas, detectar intrusos en videos de vigilancia, detección rápida de rostros y ayuda en la navegación de robots.

4.0.2.1 PyTorch

Es un framework multiplataforma de código abierto de aprendizaje automático desarrollado principalmente por Meta (antiguamente conocida como Facebook). Su funcionamiento y desarrollo de sus componentes se centra en una clase llamada Tensor, un array multidimensional optimizado para utilizarse tanto en cálculos en CPU o GPU con lo que obtiene un alto rendimiento computacional.

La plataforma permite la creación de redes neuronales desde cero, teniendo control absoluto de la arquitectura de la red y la composición de las capas, y ofrece un entorno de prueba con el que comparar resultados sea sencillo.

También ofrece redes ya probadas preparadas para entrenar (como Resnet o YOLO) y modelos ya entrenados para funciones específicas, como la transformación de texto a habla y viceversa o un generador de caras humanas.

4.0.2.2 YOLO

You Only Look Once (YOLO) Bochkovskiy y cols. (2020), ? es un algoritmo de detección de objetos que pretende detectar la posición y la clase de los posibles elementos que aparecen en una imagen. En lugar de intentar realizar predicciones en distintos rangos de píxeles del

fotograma, funciona pasando la imagen completa por una CNN que nos devuelve un listado de los objetos encontrados, junto a las coordenadas relativas en donde se encuentran y el grado de fiabilidad que tiene la predicción.

En este trabajo se utilizará una de las versiones más recientes v5, publicada originalmente en el 2020. Esta implementación contiene varias versiones distintas de la red dependiendo del tamaño y cantidad de capas que contenga. A mayor tamaño se obtienen unos resultados más precisos, con la desventaja de tener un mayor coste temporal y computacional. Para el uso que daremos en este trabajo se ha optado utilizar la red pequeña (puede consultarse su arquitectura en la figura 4.1) ya que proporciona resultados razonables y nos permite utilizarla en tiempo real.

4.0.2.3 Protocolo de transmisión en tiempo real

El Protocolo de transmisión en tiempo real (RTSP) es un protocolo a nivel de aplicación para el control de la entrega de datos con propiedades en tiempo real. El RTSP proporciona un marco extensible para permitir la entrega controlada y a la carta de datos en tiempo real, como el audio y el vídeo. Las fuentes de datos pueden incluir tanto fuentes de datos en vivo como clips almacenados.

4.0.3 Hardware

En cuanto al hardware se describe la cámara empleada.

4.0.3.1 Cámara

Se utiliza una cámara conectada a la red para la obtención de datos y comprobación del funcionamiento del trabajo con las siguientes características:

- Resolución de 4MP (2560×1440)@25/30fps, máxima soportada 4MP (2688×1520)@20 fps.
- LED infrarojo incorporado con una distancia máxima de 30m
- Detección de anomalías: Detección de movimiento, manipulación de vídeo, ausencia de tarjeta SD, tarjeta SD llena, error de tarjeta SD, red desconectada, conflicto de IP, acceso ilegal, detección de tensión.
- Códec H.265, alta relación de compresión, tasa de bits ultrabaja

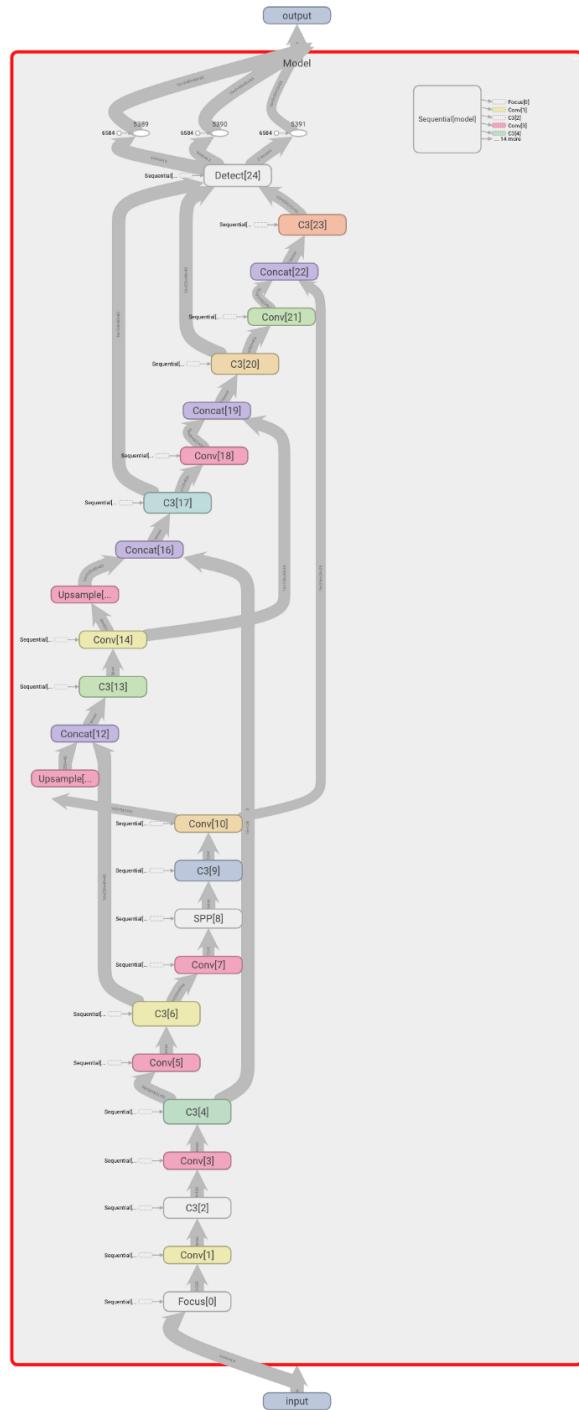


Figura 4.1: Arquitectura de YOLOv5s.

5 Desarrollo y resultados

El proyecto está compuesto por dos partes especializadas para su funcionamiento: un detector de aves y un generador de trayectorias, que siguen el siguiente flujo de trabajo (figura 5.1) que será explicado a continuación con más detalle.

1. Detector de aves.

Apertura del flujo de vídeo.

Procesamiento de fotogramas usando YOLOv5.

Si se detecta un ave se guardan las imágenes, si no, se vuelve al paso anterior.

2. Generador de trayectorias.

Se carga en memoria las imágenes a procesar.

Se aplica un algoritmo de extracción de fondo a cada una.

Las imágenes se combinan con un algoritmo de mezcla líneal.

5.1 Detector de aves

Nuestro programa comienza conectándose mediante una implementación de RTSP proveniente de OpenCV a nuestra cámara estática colocada en el parque natural, almacenando en memoria temporal los fotogramas de aproximadamente dos segundos vídeo (considerando el framerate de la cámara disponible, son entre 40-60 fotogramas).

Cuando se llega a ese límite, se envían las imágenes para ser procesadas por YOLOv5, el detector de objetos, que dará como resultado un listado de las clases encontradas en cada

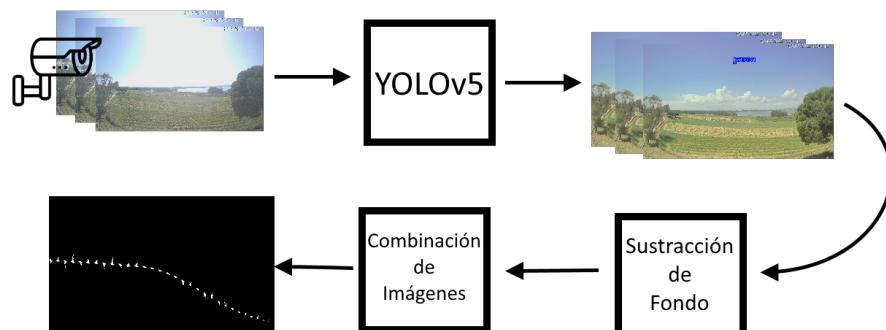


Figura 5.1: Flujo de trabajo del proyecto.



Figura 5.2: Pájaro detectado en tabla 5.1.

imagen. Esta lista contiene una línea por cada clase encontrada en la que nos indica las coordenadas relativas en la imagen donde se encuentra (xy de la esquina superior izquierda y xy de la esquina inferior derecha), el grado de confianza sobre su predicción (donde 1 significa el 100%) y la clase en la que ha clasificado el elemento. Puede verse un ejemplo de los datos obtenidos en la tabla 5.1 y uno de una imagen etiquetada en la figura 5.3.

En el momento que se detecta un ave, las imágenes se guardan en disco, junto a las coordenadas calculadas para continuar con su procesamiento más adelante. Si no se detecta nada, se descartan los fotogramas.

xmin	ymin	xmax	ymax	confidence	name
377.36	379.6	403.74	405.68	0.32	bird

Tabla 5.1: Resultado de la detección en figura 5.2.

5.1.1 Limitaciones y problemas

Durante el desarrollo y pruebas del programa han surgido tres problemas principales que han limitado el uso del detector y ha requerido una serie de compromisos funcionales. La cámara que hemos utilizado cambia automáticamente su forma de grabar a luz infrarroja en el momento que anocchece, reduciendo drásticamente la efectividad del detector. Esto si se combina con condiciones climáticas adversas, obtenemos situaciones en las que directamente se anula nuestra capacidad de detección, como se puede ver en la figura 5.4, por lo que se ha evitado usar datos nocturnos.



Figura 5.3: Pájaro detectado correctamente.

Por otro lado nos podemos encontrar en la situación en la que el detector simplemente falle y clasifique el ave como otra clase (figura 5.5), normalmente como una cometa o un avión en nuestras pruebas, o que no consiga detectar nada (figura 5.6). Para limitar los problemas que ocasiona perder estos datos en las trayectorias, se conservan los fotogramas que rodeen cualquier detección positiva de un ave. Adicionalmente se ha decidido combinar clases concretas (avión y cometa) a la clase *pájaro* para reducir la cantidad de falsos negativos obtenidos.

5.2 Generador de trayectorias

Para el tratamiento de las trayectorias inicialmente se propuso un sistema que calculara una aproximación de la línea seguida por el pájaro a partir de los puntos de posición obtenidos por nuestro detector como se puede ver en la tabla 5.1. Por desgracia, esta solución falla en el momento que no se detecte el ave, dando como resultado trayectorias inconsistentes o demasiado dispares a la realidad. Para solventar este tipo de problemas, generamos las trayectorias directamente a partir de las imágenes utilizando un algoritmo de sustracción de fondo ya implementado en OpenCV para aislar la silueta de las aves. Posteriormente se combinan con un algoritmo de mezcla lineal y se recorta usando las posiciones de inicio y fin de la primera y última detección para obtener como resultado final una imagen que describe la trayectoria completa (figuras 5.7 y 5.8). Una gran ventaja de esta solución es que al poder ver la silueta del ave, conservamos información extra que puede ayudar a su clasificación, como su forma y su aleteo.



Figura 5.4: Lluvia durante la noche.



Figura 5.5: Pájaro detectado como otra clase.



Figura 5.6: Pájaro no detectado.

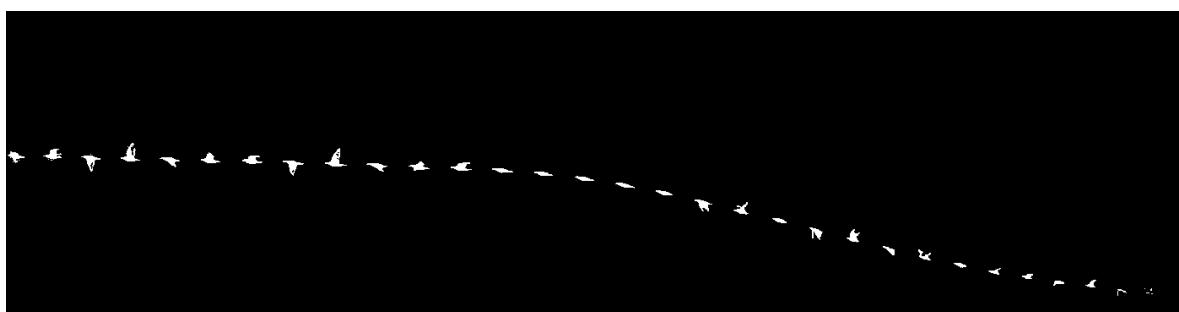


Figura 5.7: Trayectoria detectada de un ave.

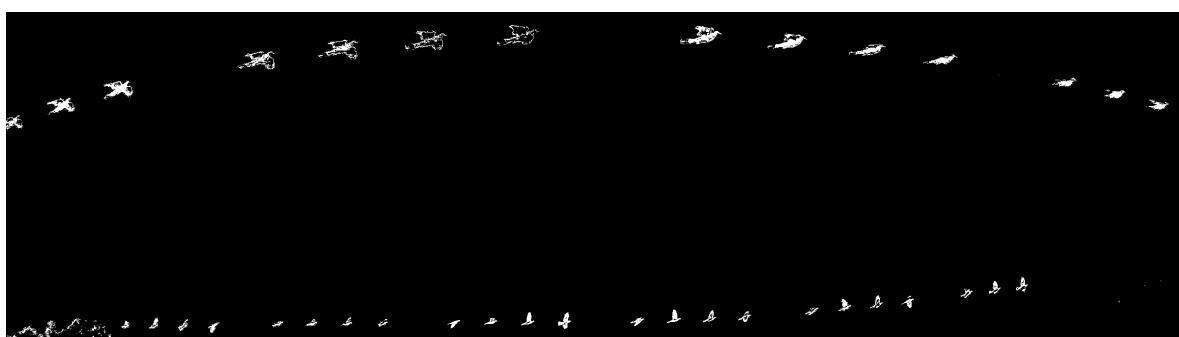


Figura 5.8: Trayectoria detectada de dos aves.

5.3 Dataset

Este conjunto de datos estará formado por las imágenes resultantes del paso anterior (ver figura 5.7) debidamente etiquetadas con la especie a la que pertenece. Este es un proceso manual que tiene que ser realizando por un experto que pueda reconocer las especies autóctonas de la zona. Siendo necesario realizar mucha más experimentación con estos datos, es altamente probable que requieran de procesamientos adicionales para ser más efectivos. Por ejemplo, en la figura 5.8 tenemos la trayectoria de dos aves que podrían ser de especies distintas y la imagen debería de ser dividida para utilizarse en el sistema.

6 Conclusiones

Con este trabajo se emprenden los primeros pasos para la investigación en el seguimiento y análisis de la trayectoria de las aves. Se ha implementado una aplicación basada en técnicas de visión y aprendizaje profundo que almacena detecciones de aves desde un flujo de vídeo en tiempo real del Parque natural de las Lagunas de La Mata y Torrevieja, desde las cuales generamos unas imágenes representativas de la trayectoria tomada por el ave.

A partir de ellas se pretende la generación de un nuevo dataset que incluirá la trayectoria etiquetada con el tipo de ave que aparece en las imágenes.

6.1 Ampliaciones futuras

A causa del limitado tiempo de desarrollo han quedado varias funcionalidades fuera del ámbito de este trabajo que no ha sido posible implementar.

La más importante es la unión de todos los procesos para diseñar un flujo de trabajo completamente automático. Ahora mismo se sigue requiriendo de intervención humana para la generación de las imágenes de trayectoria.

Además, la creación del dataset es un proceso altamente manual ya que requiere la identificación y etiquetado de cada imagen por parte de un experto haciendo de esta una tarea muy lenta y tediosa. Agregar un paso más al flujo de trabajo, combinando con el funcionamiento de otros trabajos de reconocimiento de aves con mayor abundancia de datos podría utilizarse para automatizar este etiquetado. Por ejemplo, si se sincronizara el sistema de generación de trayectorias propuesto junto con otro que reconociera el ave a partir de una fotografía del espécimen, el trabajo de experto pasaría a ser de validación en lugar de reconocimiento y etiquetado.

Como ampliación final, una vez que se haya acumulado suficiente información, quedaría pendiente la implementación y experimentación de una nueva red encargada de aprender nuestro dataset.

Bibliografía

- Akçay, H. G., Kabasakal, B., Aksu, D., Demir, N., Öz, M., y Erdoğan, A. (2020, 7). Automated bird counting with deep learning for regional bird distribution mapping. *Animals*, 10, 1-24. doi: 10.3390/ani10071207
- Benjumea, A., Teeti, I., Cuzzolin, F., y Bradley, A. (2021). *Yolo-z: Improving small object detection in yolov5 for autonomous vehicles*. arXiv. Descargado de <https://arxiv.org/abs/2112.11798> doi: 10.48550/ARXIV.2112.11798
- Bochkovskiy, A., Wang, C., y Liao, H. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *CoRR, abs/2004.10934*. Descargado de <https://arxiv.org/abs/2004.10934>
- Brodrick, P. G., Davies, A. B., y Asner, G. P. (2019, 8). *Uncovering ecological patterns with convolutional neural networks* (Vol. 34). Elsevier Ltd. doi: 10.1016/j.tree.2019.03.006
- Cooke, S. J., Hinch, S. G., Wikelski, M., Andrews, R. D., Kuchel, L. J., Wolcott, T. G., y Butler, P. J. (2004). Biotelemetry: a mechanistic approach to ecology. *Trends in Ecology & Evolution*, 19(6), 334-343. Descargado de <https://www.sciencedirect.com/science/article/pii/S0169534704001120> doi: <https://doi.org/10.1016/j.tree.2004.04.003>
- Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., y Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: Optimizing animal ecology with deep learning. *Frontiers in Marine Science*, 7. Descargado de <https://www.frontiersin.org/article/10.3389/fmars.2020.00429> doi: 10.3389/fmars.2020.00429
- Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep learning*. MIT Press. (<http://www.deeplearningbook.org>)
- Huang, Y.-P., y Basanta, H. (2019). Bird image retrieval and recognition using a deep learning platform. *IEEE Access*, 7, 66980-66989. doi: 10.1109/ACCESS.2019.2918274
- Kahl, S., Wood, C. M., Eibl, M., y Klinck, H. (2021a). Birdnet: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61, 101236. Descargado de <https://www.sciencedirect.com/science/article/pii/S1574954121000273> doi: <https://doi.org/10.1016/j.ecoinf.2021.101236>
- Kahl, S., Wood, C. M., Eibl, M., y Klinck, H. (2021b, 3). Birdnet: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61. doi: 10.1016/j.ecoinf.2021.101236
- Schulzrinne, Columbia, Rao, Netscape, Lanphier, y RealNetworks. (1998). *Real Time Streaming Protocol (RTSP)*. <https://www.ietf.org/rfc/rfc2326.txt>.

Takeki, A., Trinh, T., Yoshihashi, R., Rei, K., Iida, M., y Naemura, T. (2016, 12). Combining deep features for object detection at various scales: Finding small birds in landscape images. *IPSJ Transactions on Computer Vision and Applications*, 8. doi: 10.1186/s41074-016-0006-z

Takeki, A., Trinh, T. T., Yoshihashi, R., Kawakami, R., Iida, M., y Naemura, T. (2016). Detection of small birds in large images by combining a deep detector with semantic segmentation. *2016 IEEE International Conference on Image Processing (ICIP)*, 3977-3981.