DETECCIÓN Y CLASIFICACIÓN DE HUEVOS DE PARÁSITOS EN IMÁGENES MICROSCÓPICAS

Anibal Pedraza, Jesus Ruiz-Santaquiteria, Oscar Deniz y Gloria Bueno
VISILAB, ETSII, Avda Camilo Jose Cela, 13071, Ciudad Real, España
{anibal.pedraza, jesus.ralegre, oscar.deniz, gloria.bueno}@uclm.es

Resumen

Las afecciones por parásitos intestinales son un grave problema de salud con un alto impacto en algunas áreas geográficas. Dado que actualmente la evaluación de estas enfermedades se realiza de forma manual a través de expertos, es posible aplicar técnicas de aprendizaje automático para ayudar en el desarrollo de este tarea, reduciendo al menos la carga de trabajo. Esto podría llevar a un menor tiempo de detección de la enfermedad y a la aplicación de un tratamiento adecuado más rápidamente. En el contexto del aprendizaje profundo, se han propuesto muchas técnicas de detección de objetos, validadas en conjuntos de datos de propósito general, como ImageNet o COCO. En este trabajo, proponemos una unión de varias de ellas, incluyendo técnicas recientes basadas en Transformers, para afrontar esta tarea particular. Como resultado, la unión de los métodos TOOD, Cascade-RCNN (Swin-Transformer), Cascade-RCNN (ConvNeXt) y YOLOX, aplicados a la detección de este tipo de imágenes, consigue un valor de 0,915 para la métrica Intersección sobre la Unión (Intersection over Union, IoU), el cual es mayor que los resultados obtenidos por cada uno de los métodos por separado.

Palabras clave: Huevos de parásitos, Detección de Objetos, Redes Neuronales, Aprendizaje Profundo, Transformers.

1. Introducción

Las infecciones por parásitos intestinales son una de las principales causas de mortalidad hoy en día, con un mayor impacto en áreas tropicales, donde las temperaturas son más altas. Según la Organización Mundial de la Salud (OMS), 1.500 millones de personas (un 24 % de la población mundial) están infectadas por Geohelmintiasis (infecciones de Helminth transmitidas a través del suelo). De este grupo, un subconjunto de 838 millones son niños que requirieron quimioterapia por esta causa. La mayoría de estas infecciones pueden causar síntomas como la diarrea, o incluso peores como anemia y malnutrición. Como consecuencia, esto puede derivar en niños que sufrirán de trastornos del crecimiento. La mayoría de las personas infectadas pueden vivir normalmente durante su incubación, lo que causa la transmisión de estos parásitos a otros individuos de su comunidad. En los últimos años, ha habido mejoras en estas regiones, por ejemplo en higiene personal, sanidad y educación. Esto ha inducido a la reducción en parte de estas helmintiasis. Sin embargo, la enfermedad puede manifestarse también en pacientes asintomáticos, induciéndolos a padecer una enfermedad crónica. En otros países en vías de desarrollo, los protozoos intestinales han sido reconocidos también como una importante causa de distintas enfermedades.

Por tanto, la detección de huevos de parásitos se presenta como un problema relevante que afecta a la salud humana, y por la cual se pueden contraer distintas enfermedades [20]. Hay múltiples especies diferentes que son conocidas, mientras que muchas otras siguen descubriéndose hoy en día a través de la investigación biológica realizada en excavaciones arqueológicas [11]. En la mayoría de casos, este diagnóstico se hace a través de la observación directa en un laboratorio. Sin embargo, este método presente una baja sensibilidad, requiriendo la inversión de bastante tiempo (hasta 30 minutos para una única muestra en el microscopio) por parte de expertos en el laboratorio. Por todo ello, el desarrollo de sistemas automáticos para la evaluación de muestras es crítica.

En los últimos años, los sistemas de aprendizaje automático están experimentando un progreso significativo, particularmente en el campo de la visión artificial. Las recientes arquitecturas basadas en deep learning han conseguido resultados impresionantes en tareas como el reconocimiento de imágenes, la detección de objetos, la segmentación de imágenes o la generación de datos, principalmente gracias a la aparición de las Redes Neuronales Convolucionales (CNNs) y, más recientemente, de las arquitecturas basadas en Transformers.

En este trabajo se presenta la contribución de distintos métodos de detección de objetos, a través del aprendizaje automático y el aprendizaje pro-

https://doi.org/10.17979/spudc.9788497498418.1023

fundo, al problema de detección de huevos de parásitos en imágenes microscópicas.

El resto del artículo se organiza de la siguiente forma. En la Sección 2 se revisa el trabajo existente relacionado con la tarea de detección automática de huevos de parásitos. El conjunto de datos utilizado se describe brevemente en la Sección 3. En la Sección 4 se describen los métodos utilizados. Los experimentos realizados y los resultados obtenidos se resumen en la Sección 5. Finalmente, las conclusiones y el trabajo futuro se presentan en la Sección 6.

2. Antecedentes

En el estado del arte es posible encontrar varios trabajos que cubren este tema desde el campo del aprendizaje profundo. Por ejemplo, en [18] se desarrolla un método en dos pasos. En primer lugar, una Red Generativa Adversaria (GAN), denominada Pix2Pix, se utiliza para mejorar la calidad visual de las muestras a través de la técnica de super-resolución. Posteriormente, se aplica una detección a través de una red Faster-RCNN para obtener una precisión del 0,97 de media para las 5 clases consideradas. En el trabajo de Suwannaphong et al. [22] se emplean dos redes habituales en el estado del arte, como son AlexNet y ResNet50, de tal forma que se comprueba la diferencia entre los resultados obtenidos por imágenes en baja resolución de un microscopio de bajo coste, frente a imágenes de mayor resolución obtenidas con un equipo más caro. En este caso, se emplean 4 clases, obteniendo un 98,25 % de exactitud para Resnet50 (y un 96,93 % para AlexNet). El método empleado se basa en una ventana deslizante, de tal forma que la imagen se evalúa en múltiples subregiones. Otros trabajos reducen aún más el número de clases, centrándose un una única categoría. Este es el caso del trabajo realizado en [1], el cual emplea un red neuronal con distintas capas personalizadas para maximizar el rendimiento de la clase "Ascaris lumbricoides" en tres etapas de su ciclo vital: infertilidad, fertilidad y decorticación. Para esta tarea, obtienen un 93,33 % de exactitud. Trabajos recientes, como en [13], también intentan utilizar la combinación de diferentes métodos de aprendizaje profundo. En concreto, emplean los métodos Faster-RCNN, RetinaNet y CenterNet, a través de la librería Detectron2, obteniendo como máximo una Precisión media (mean Average Precision, mAP) del 0,74.

3. Conjuntos de datos

Para este trabajo, se ha empleado el complejo conjunto de datos denominado "Chula-ParasiteEgg-11" [19], el cual contiene un total de 11 clases de huevos de parásitos extraídos del frotis de muestras fecales, con tamaños aproximados de entre 15 y 100 micras. Las imágenes han sido adquiridas utilizando multitud de dispositivos diferentes, incluyendo cámaras como la Canon EOS 70D, a través de un microscopio Olympus BX53, o una cámara DS-Fi2 Nikon junto al microscopio Nikon Eclipse Ni. Además también se han utilizado dispositivos móviles como el Samsung Galaxy J7 iPhone 12 y iPhone 13 con lentes de aumento 10x. Como resultado, se tiene una gran variedad de condiciones de resolución, luminosidad y calidad. Por ejemplo, algunas imágenes están desenfocadas, otras tienen algún tipo de ruido e incluso también esta presente el efecto motion blur, el cual distorsiona la imagen cuando se ha capturado en movimiento a través de la pletina motorizada del microscopio. Esta gran variedad de calidad y configuraciones de las imágenes permite extraer un conjunto muy variado de características. El objetivo es, por tanto, que esta variabilidad ayude a la creación de modelos más robustos. En particular, este conjunto de datos cuenta con un total de 11.000 imágenes para entrenamiento (1.000 imágenes de cada clase) y 2.200 para test.

En la Tabla 1 se muestra un ejemplo de cada una de las clases del conjunto de datos. Con el objetivo de observar el aspecto de cada especie con mayor detalle, se han recortado los objetos de interés de la imagen completa. En la Figura 1 pueden observarse algunas muestras completas. Nótese que, en cada una, suele aparecer únicamente un solo huevo de parásito. Sin embargo, en el dataset completo hay algunas excepciones donde aparecen varios en la misma imagen. Por tanto, tendrá que determinarse algún mecanismo para la posible detección de objetos solapados.

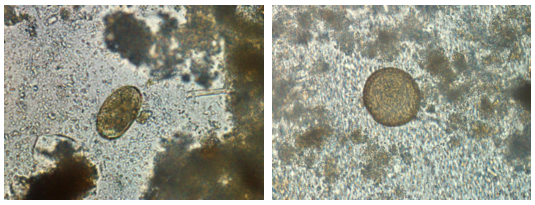

Figura 1: Ejemplos de muestras completas del conjunto de datos

4. Métodos

En este trabajo se han considerado diferentes métodos de detección de objetos con aprendizaje profundo. El primero de ellos es Faster-RCNN [21], cuyo objetivo es reducir el tiempo computacional dedicado a la propuesta de regiones candidatas (en comparación al método original RCNN (Region

https://doi.org/10.17979/spudc.9788497498418.1023

Tabla 1: Ejemplos de cada clase del dataset

Clase	Imagen	Clase	Imagen
Ascaris lumbricoides		Capillaria philippinensis	
Enterobius vermicularis		Fasciolopsis buski	
Hookworm egg		Hymenolepis diminuta	
Hymenolepis nana		Opisthorchis viverrine	
Paragonimus spp		Taenia spp egg	
Trichuris trichiura			

Based Convolutional Neural Network). Este método define una Red de Proposición de Regiones (Region Proposal Network, RPN), la cual comparte las características previamente aprendidas por la red de detección, permitiendo proponer las regiones con un menor coste computacional. Esta RPN es una red convolucional completamente conectada que predice simultáneamente la localización de regiones candidatas y una puntuación en cada posición sobre la posibilidad de que estas contengan un objeto. Por tanto, la RPN se entrena desde cero para generar propuestas de regiones de alta calidad, las cuales se usan en el resto de la arquitectura RCNN para llevar a cabo la detección en si. En consecuencia, la Faster-RCNN surge de la unión de una RPN y una RCNN en una única red, compartiendo parte de los pesos de sus capas convolucionales, en lo que en la terminología reciente sobre redes neuronales se viene denominando como mecanismos de "atención", siendo la RPN el componente que le indica a la red unificada donde debe observar. Específicamente, en este trabajo se emplea una arquitectura basada en una red Resnet50-FPN como base, utilizando pesos pre-entrenados de COCO.

Algunas variantes de Faster-RCNN como Mask-RCNN [12] y Cascade Mask-RCNN [2], también se han tenido en cuenta. Mask-RCNN es una modificación de Faster-RCNN que incluye una rama adicional para la segmentación de las instancias detectadas. Por otro lado, Cascade Mask-RCNN propone una arquitectura de detección a diferentes escalas, compuesta de una secuencia de detectores entrenados con diferentes umbrales de IoU.

Otro de los métodos probados es TOOD [8]. Habitualmente, la detección de objetos en una etapa se implementa optimizando dos sub-tareas: loca-

lización y clasificación de objetos, usando dos cabeceras de red en ramas paralelas del grafo de la red neuronal. Sin embargo, esto puede conducir a un cierto nivel de desalineamiento espacial entre las predicciones de ambas tareas. En este método, se propone la Detección de Objetos en una etapa alineado con la Tarea (Task-aligned One-stage Object Detection, TOOD), que explícitamente alinea las dos tareas en la etapa de aprendizaje. En primer lugar, la novedosa Cabecera Alineada a la Tarea (Task-aligned Head, T-Head) ofrece el mejor balance entre la interacción de las tareas para el aprendizaje de características específicas de cada una, así como una mayor flexibilidad para aprender este alineamiento a través de un predictor alineado a la tarea. En segundo lugar, el Aprendizaje Alineado a la Tarea (Task Alignment Learning, TAL) explícitamente acerca (o incluso unifica) los puntos de detección propuestos por las dos tareas durante el entrenamiento, a través del diseño de un esquema orientado a la asignación de candidatos y una función de pérdida alineada a la tarea. Para este trabajo, la arquitectura empleada como base es una Resnet101-FPN con convoluciones deformables.

Una variante de la conocida familia de métodos "You Only Look Once" (YOLO), YOLOX [9], también se ha probado. Este presenta algunas mejoras sobre la serie de métodos originales, obteniendo un nuevo detector de alto rendimiento. Con este método, se transforma la red YOLO hacia un estilo libre de puntos de anclaje, aplicando otras técnicas avanzadas de detección, como un desdoblamiento entre la cabecera y la estrategia de asignación de etiquetas, alcanzando resultados comparables al estado del arte en una gran variedad de modelos y conjuntos de de datos a gran escala. En este trabajo se ha empleado la arquitectura más grande disponible (versión "x" del modelo), con una red Darknet53 como base, pre-entrenada con pesos de COCO.

Finalmente, se han analizado recientes arquitecturas basadas en Transformers para resolver este problema, siendo el primer trabajo (en conocimiento de los autores) que utiliza este método aplicado a este problema. El creciente interés por este tipo de arquitecturas para la resolución de tareas de Procesamiento del Lenguaje Natural (NLP) ha impulsado su aplicación en problemas de visión artificial como el reconocimiento de imágenes o la detección de objetos con resultados prometedores. El método original del Transformer se presentó en 2017 y se basa en mecanismos de auto-atención [23]. La parte más relevante de esta arquitectura es la rama del codificador, que básicamente procesa secuencias de datos de entrada (por ejemplo, una secuencia de palabras) para generar una representación de características enriquecida que contiene información contextual de todos los componentes de la secuencia.

En Dosovitskiy *et al.* [7] se propuso una adaptación sencilla de esta arquitectura para la clasificación de imágenes, el "Vision-Transformer" (ViT). En este método, la imagen de entrada se divide en parches de tamaño fijo (16x16 en este caso) y luego pasa a través de varias capas de auto-atención. Después, la representación de características generada es la entrada de una red neuronal completamente conectada que realiza la clasificación final de la imagen. Este método es capaz de obtener un rendimiento equivalente al estado del arte en varios conjuntos de datos de referencia sin necesidad de utilizar CNNs para la extracción de características.

Los Transformers también se han aplicado a tareas de detección de objetos. En Carion *et al.* [3] se propuso una arquitectura codificador-decodificador en la que la detección es manejada como un problema de predicción basado en conjuntos, también llamado "Detection Transformer"(DETR). Dado un conjunto fijo de regiones candidatas, DETR genera un conjunto de predicciones teniendo en cuenta el contexto de la imagen global extraído del codificador del Transformer y las regiones candidatas. De la misma forma que en el método ViT, la imagen de entrada se divide en parches de tamaño fijo pero en este caso la imagen se pasa previamente por una CNN (ResNet) antes del bloque de codificación. Este método muestra un rendimiento similar a los mejores métodos de detección en el popular conjunto de datos de detección de objetos COCO.

Si bien DETR es en sí mismo un método de detección de objetos y segmentación, ViT y otras arquitecturas basadas en bloques de codificación de Transformers se pueden utilizar como métodos de extracción de características ("backbones") para la mayoría de los métodos de detección y segmentación de objetos anteriormente comentados, como Faster-RCNN, Mask-RCNN o Cascade Mask-RCNN. En Liu *et al.* [14] se propuso un nuevo método basado en Transformers que se puede aplicar como extractor de características para aplicaciones de visión artificial. Este método tiene como objetivo resolver los problemas relacionados con la escala y la alta resolución de las imágenes en tareas de visión por computador en comparación con, por ejemplo, las secuencias de texto en el ámbito del procesamiento del lenguaje natural (NLP). Para abordar estos inconvenientes, los autores proponen una arquitectura jerárquica basada en ventanas, el Swin-Transformer. Este esquema limita el cálculo de auto-atención a ventanas

 https://doi.org/10.17979/spudc.9788497498418.1023

locales que no se superponen, al mismo tiempo que permite la conexión entre ventanas, lo que permite el modelado a múltiples escalas. En Xie *et al.* [24] también se propuso un enfoque de aprendizaje auto-supervisado que incluye Transformers como extractor de características, MoBY. Este método combina dos enfoques auto-supervisados populares, MoCo v2 [6] y BYOL [10], que en combinación con Swin-Transformer permite usar las representaciones aprendidas en tareas posteriores como la detección de objetos y segmentación semántica.

5. Experimentos y Resultados

Para este trabajo, todos los experimentos se han llevado a cabo en un equipo con una GPU NVIDIA Quadro RTX 5000 16 GB. No se aplica ningún pre-procesamiento a las imágenes de entrada, únicamente se escalan y normalizan para adaptarlas a los requisitos de cada método (en función del pre-entrenamiento realizado). Sin embargo, se han incluido varias técnicas estándar de aumento de datos (por ejemplo, rotaciones aleatorias, recortes aleatorios y relleno) para permitir una mejor generalización y evitar el sobreajuste del modelo a los datos de entrenamiento.

Con respecto a los modelos pre-entrenados y los detalles de implementación, todos los extractores de características se entrenaron previamente en el conjunto de datos ImageNet-1K. Además, los métodos de detección completos también se preentrenaron en el conjunto de datos COCO. Para el entrenamiento y evaluación de los métodos Faster-RCNN, TOOD y YOLOX se ha utilizado la librería MMDetection [5]. Para el método DETR se ha probado la implementación de la librería Hugging Face [4]. Esta implementación permite el uso de modelos previamente entrenados y proporciona métodos para re-entrenar el modelo en conjuntos de datos personalizados. Los extractores de características basados en Swin-Transformer, utilizados con varios métodos de detección de objetos, se han obtenido de la implementación oficial [16]. Los modelos pre-entrenados de MoBY y ConvNeXt, también están disponibles en sus respectivos repositorios oficiales [15, 17]. El optimizador seleccionado para todas las pruebas es AdamW, que es una variante mejorada de Adam para reducir el sobreajuste del modelo. La tasa de aprendizaje inicial se adapta al número de GPU disponibles (1 en nuestro caso), estableciendo este valor inicialmente en 1e-4. Para ello se toma el valor de tasa de aprendizaje de referencia en el modelo original preentrenado, el cual se desarrolló en un entorno con 8 GPUs disponibles, dividiéndolo por ese valor para conseguir un aprendizaje proporcional con una única GPU. El número de épocas de entrenamiento y el tamaño del "batch" se ha ajustado para cada método, realizando diferentes iteraciones para cada método hasta encontrar la mejor configuración de cada uno.

Todos los métodos propuestos han sido entrenados en primer lugar de forma independiente, para posteriormente unir todas las detecciones en un único ensamble de modelos, con el objetivo de cubrir las posibles detecciones faltantes de alguno de los modelos por las propuestas por otro. Para ello, se realiza la predicción de las imágenes para cada uno de los modelos, escogiendo la detección con una mayor puntuación de confianza en los casos en que haya un solapamiento de detecciones en la misma zona de la imagen. Para ello, se establece que haya un umbral mínimo de solapamiento del 5 %. El resumen de los resultados obtenidos se puede observar en la Tabla 2. En ella se muestran las métricas mIoU y F1 score para el conjunto de test propuesto (2.200 imágenes).

El mIoU se calcula como la media del IoU de las 11 clases del conjunto de datos. Cada uno de ellos se calcula como la división entre el solapamiento y la unión de las regiones detectadas. Por otro lado, el F1-Score se calcula como el doble del solapamiento, dividido por la suma de las áreas tanto de la región detectada como de la máscara original.

En los experimentos 1-8, se observan los resultados para cada uno de los métodos comentados anteriormente, así como de sus distintas variantes. En primer lugar, mencionar que tanto DETR como Mask-RCNN obtienen unos resultados relativamente bajos, por lo que no serán considerados para su aplicación en los distintos ensambles. Cabe destacar, que el método YOLOX, por si mismo, es capaz de alcanzar resultados superiores al 0,9. En primer lugar, en el experimento 9 (Ensamble de los métodos 3, 4, 7 y 8) se prueban todos los métodos unidos, escogiendo la mejor versión de la familia de métodos Cascade. Como resultado, se obtiene un mIoU de 0,905, el cual ya es superior al obtenido por el mejor método (YOLOX), de forma individual. A continuación, en el experimento 10 (Ensamble de los métodos 7 y 8), se prueba la unión exclusivamente de los dos mejores métodos. Esto da como resultado un mIoU de 0,912, el cual es superior al ensamble anterior. Es interesante observar que la unión en este caso de un menor número de métodos da un mejor resultado. Por tanto se puede inducir que algunos métodos pueden desvirtuar o empeorar los resultados, en lugar de aportar a la unión. Finalmente, en el experimento 11 (Ensamble de los métodos 4, 6, 7 y 8) se muestra el mejor resultado que ha sido posible conseguir. Para ello se han realizado distintas pruebas, eliminando alguno de los métodos

Tabla 2: Resumen de resultados

ID	Método	Backbone	mIoU	F1score
1	DETR	Resnet50-FPN	0,656	0,854
2	Mask-RCNN	Swin-Transformer-Small	0,708	0,862
3	Faster-RCNN	Resnet50-FPN	0,817	0,923
4	TOOD	Resnet101-FPN	0,844	0,935
5	Cascade Mask-RCNN	Swin-Transformer-MoBY-Tiny	0,848	0,940
6	Cascade Mask-RCNN	Swin-Transformer-Base	0,875	0,955
7	Cascade Mask-RCNN	ConvNeXt-Large	0,880	0,955
8	YOLOX	Darknet53	0,901	0,969
9	Ensamble (3+4+7+8)	-	0,905	0,967
10	Ensamble (7+8)	-	0,912	0,971
11	Ensamble (4+6+7+8)	-	0,915	0,974

que menos aportaban en el ensamble anterior (como era el Faster-RCNN) y añadiendo otros que han aportado variabilidad a las detecciones (como es la variante Swin-Transformer del Cascade Mask-RCNN). Este último es importante, dado que al pertenecer a una familia de arquitecturas completamente distinta a todas las demás (ya que está basada en transformers, con mecanismos de atención, como ya se ha mencionado), aporta una mayor variabilidad en la forma de detectar los objetos. Esto es especialmente importante en problemas como este, donde hay muchos tipos distintos de imagen y naturaleza de los objetos. El resultado obtenido por este ensamble, con un mIoU de 0,915 aumenta un 1 % los resultados obtenidos de forma aislada, lo cual es relevante en problemas con un número de imágenes tan elevado.

Finalmente, la Figura 2 muestra algunos ejemplos de detección para cada clase presente en el conjunto de datos.

6. Conclusiones

En este trabajo se ha presentado un método de unión de modelos de detección para la tarea de clasificación de imágenes de huevos de parásitos. Una de las consideraciones más importantes que se pueden extraer es cómo las detecciones de varios modelos, con enfoques y estrategias distintas, pueden combinarse para mejorar el rendimiento que obtendrían cada uno por separado. Como resultado, se ha comprobado que estas predicciones pueden ser complementarias. Esto puede explicarse dado que cada una de las arquitecturas pueden funcionar mejor para distintas circunstancias: objetos de mayor o menor tamaño, imágenes con un alto contraste o con zonas borrosas e incluso la robustez frente a la presencia de artefactos en la imagen.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto TIN2017-82113-C2-2-R del Ministerio de Economía y Competitividad de España, el proyecto DISARM (PDC2021-121197) financiado por el MCIN/AEI/ 10.13039/501100011033 y "European Union NextGenerationEU/PRTR" y el proyecto SBPLY/17/180501/000543 y SBPLY/21/180501/000025 de la Junta de Comunidades de Castilla-La Mancha; así como por los Contratos Predoctorales de Formación FPU17/04758 y PRE2018-083772 del Ministerio de Ciencia, Innovación y Universidades de España.

English summary

DETECTION AND CLASSIFICATION OF PARASITIC EGG IN MICROSCOPY IMAGES

Abstract

Intestinal parasitic infections are a healthcare problem with a high impact in some areas. While nowadays the assessment performed by experts is mostly manual, it is possible to introduce machine learning techniques to help automating this task, or at least reduce the workload. This can lead to shorter detection times and faster treatment application. In the context of deep learning, several object detection techniques have been proposed and validated on general purpose datasets such as ImageNet or COCO. In this work, an ensemble of these, including recent Transformer-based techniques, is proposed for this particular task. The merged detections of TOOD,

 https://doi.org/10.17979/spudc.9788497498418.1023

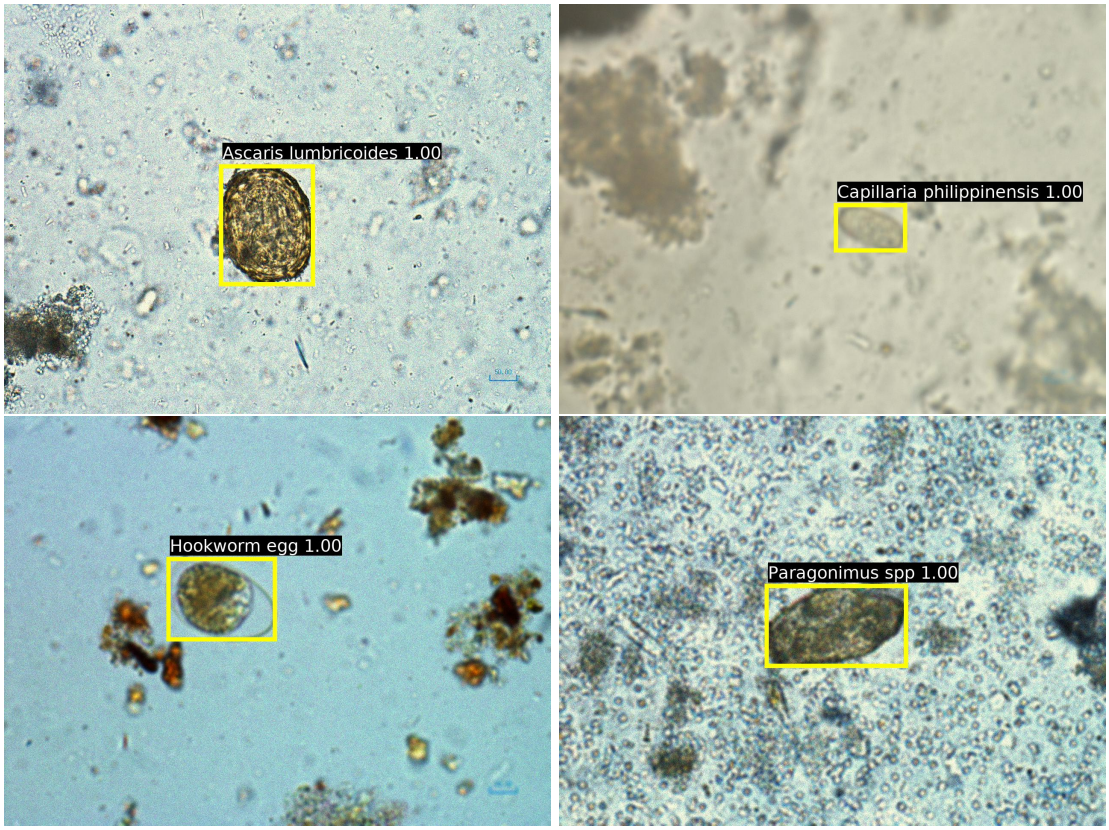

Figura 2: Ejemplos de detección y clasificación de huevos de diferentes clases

Cascade-RCNN (Swin-Transformer), Cascade-RCNN (ConvNeXt) and YOLOX applied to this problem achieved 0.915 for the Intersection over Union metric (IoU), which is larger than the result that each method obtained independently.

Keywords: Parasitic Eggs, Object Detection, Neural networks, Deep Learning, Transformers.

Referencias

[1] Butploy, N., Kanarkard, W. & Maleewong Intapan, P. (2021). Deep Learning Approach for Ascaris lumbricoides Parasite Egg Classification. *Journal of Parasitology Research, 2021.*

[2] Cai, Z. & Vasconcelos, N. (2018). Cascade R-CNN: Delving into High Quality Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6154-6162.

[3] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. & Zagoruyko, S. (2020a). End-to-End Object Detection with Transformers. *European conference on computer vision*, 213-229.

[4] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. & Zagoruyko, S. (2020b). End-to-End Object Detection with Transformers. *CoRR, abs/2005.12872.* https://arxiv.org/abs/2005.12872

[5] Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., ... Lin, D. (2019). MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv preprint arXiv:1906.07155.*

[6] Chen, X., Fan, H., Girshick, R. & He, K. (2020). Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297.*

[7] Dosovitskiy, A., Beyer, L. & Kolesnikov, A. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929.*

[8] Feng, C., Zhong, Y., Gao, Y., Scott, M. R. & Huang, W. (2021). Tood: Task-aligned one-stage object detection. *2021 IEEE/CVF In-*

ternational Conference on Computer Vision (ICCV), 3490-3499.

[9] Ge, Z., Liu, S., Wang, F., Li, Z. & Sun, J. (2021). Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*.

[10] Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z. & Gheshlaghi Azar, M. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, *33*, 21271-21284.

[11] Han, E.-T., Guk, S.-M., Kim, J.-L., Jeong, H.-J., Kim, S.-N. & Chai, J.-Y. (2003). Detection of parasite eggs from archaeological excavations in the Republic of Korea. *Memórias do Instituto Oswaldo Cruz*, *98*, 123-126.

[12] He, K., Gkioxari, G., Dollár, P. & Girshick, R. (2017). Mask R-CNN. *Proceedings of the IEEE international Conference on Computer Vision*, 2961-2969.

[13] Kitvimonrat, A., Hongcharoen, N., Marukatat, S. & Watcharabutsarakham, S. (2020). Automatic Detection and Characterization of Parasite Eggs using Deep Learning Methods. *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 153-156.

[14] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. & Guo, B. (2021). Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012-10022.

[15] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. & Guo, B. (2022a). Official implementation for Self-Supervised Learning with Swin Transformers [Accessed: 22/04/2022].

[16] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. & Guo, B. (2022b). Official implementation for Swin Transformer: Hierarchical Vision Transformer using Shifted Windows [Accessed: 22/04/2022].

[17] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T. & Xie, S. (2022). Official implementation for ConvNeXt models [Accessed: 22/04/2022].

[18] Mayo, P., Anantrasirichai, N., Chalidabhongse, T. H., Palasuwan, D. & Achim, A. (2022). Detection of Parasitic Eggs from Microscopy Images and the emergence of a new dataset. *arXiv preprint arXiv:2203.02940*.

[19] Palasuwan, D., Naruenatthanaset, K., Kobchaisawat, T., Chalidabhongse, T. H., Nunthanasup, N., Boonpeng, K. & Anantrasirichai, N. (2022). Parasitic Egg Detection and Classification in Microscopic Images [Accessed: 18/07/2022]. https://doi.org/10.21227/vyh8-4h71

[20] Peixinho, A., Martins, S., Vargas, J., Falcao, A., Gomes, J. & Suzuki, C. (2015). Diagnosis of human intestinal parasites by deep learning. *Computational Vision and Medical Image Processing V: Proceedings of the 5th Eccomas Thematic Conference on Computational Vision and Medical Image Processing (VipIMAGE 2015, Tenerife, Spain*, 107.

[21] Ren, S., He, K., Girshick, R. & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, *28*.

[22] Suwannaphong, T., Chavana, S., Tongsom, S., Palasuwan, D., Chalidabhongse, T. H. & Anantrasirichai, N. (2021). Parasitic Egg Detection and Classification in Low-cost Microscopic Images using Transfer Learning. *arXiv preprint arXiv:2107.00968*.

[23] Vaswani, A., Shazeer, N. & Parmar, N. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

[24] Xie, Z., Lin, Y., Yao, Z., Zhang, Z., Dai, Q., Cao, Y. & Hu, H. (2021). Self-Supervised Learning with Swin Transformers. *arXiv preprint arXiv:2105.04553*.

 https://doi.org/10.17979/spudc.9788497498418.1023