Comparación de los Algoritmos en Aprendizaje-Máquina Yolo v3, Faster RCNN y SSD para Detección de Objetos

Jonathan Iván Gutiérrez Gayosso[1], Dr. Luis Carlos Méndez González[2], Dr. Luis Alberto Rodríguez Picón[3], Dr. Iván Juan Carlos Pérez Olguín[4]

***Resumen*—** Desde hace décadas el estudio de la inteligencia artificial ha estado en constante desarrollo, se han logrado grandes avances en distintos campos de estudio. Este auge ha dado lugar a una gran cantidad de algoritmos que hacen uso del aprendizaje automático. Este proyecto busca comparar el rendimiento de diferentes algoritmos de visión artificial con el objetivo de determinar cuál modelo es el más optimizado para nuestros propósitos. Para lograrlo se eligieron 3 de los principales modelos de detección de objetos que se utilizan en el aprendizaje automático. Los modelos son entrenados con los mismos datos para lograr detectar diversos objetos, posteriormente se realizan distintas pruebas en donde se evalúa la detección de distintos objetivos. Con los datos recopilados se realiza una comparación en base a la precisión con la que detectaron los objetivos y el tiempo de respuesta.

***Palabras clave*—** aprendizaje automático, aprendizaje profundo, visión artificial, detección de objetos.

Introducción

La visión por computadora se trata de un campo de la inteligencia artificial que busca imitar distintas características de la visión humana. El software y el hardware informático se utilizan para analizar y procesar información y datos visuales. Incluyendo el proceso de adquirir, transmitir, filtrar, almacenar y comprender información visual. A partir de la imagen o secuencia de imágenes, se adquiere el conocimiento y la comprensión del mundo externo y se recopila la información relevante del objeto[1]. La visión por computadora asistida por sistemas de inteligencia artificial puede tener un gran impacto en procesos industriales, de manera que los costos de operación se reduzcan, la calidad de los productos mejore y se obtenga una mayor rapidez y precisión llegando al punto de poder inspeccionar una mayor cantidad de productos. La flexibilidad del sistema está directamente relacionada con los datos de entrenamiento que le suministremos al algoritmo por lo que podemos ajustarlo de acuerdo con nuestras necesidades siempre que contemos con información suficiente[2].

Este proyecto se encuentra enfocado en la comparación de 3 modelos de visión artificial que se especializan en la detección de objetos: YoloV3, Faster R-CNN y SSD. La comparación tiene como objetivo determinar cual de los 3 modelos tiene un mejor desempeño, para lograrlo todos los algoritmos serán entrenados con el mismo conjunto de datos para posteriormente ejecutarlos en distintas imágenes de prueba. El desempeño se evaluará de acuerdo con el tiempo de ejecución de cada modelo y la precisión con la que detectan distintos objetos.

Descripción del Método

Los sistemas de clasificación por visión humana presentan algunos problemas importantes. Las personas que trabajan en estos procesos pueden cometer diversos errores influenciados por su situación personal o debido a las variables del entorno en donde desempeñan sus tareas[3]. La fatiga, el estado de ánimo, las condiciones de luz o la comodidad en el trabajo son ejemplos de los factores que pueden tener un impacto en la calidad del proceso. Debido a los factores anteriores el proceso se puede volver lento y/o impreciso por lo que sería necesario aplicar medidas como el aumento de la mano de obra lo que ocasionaría mayores gastos en nuestros procesos.

Los sistemas de visión artificial pueden solventar los problemas que involucran el uso de la visión humana. Estos sistemas se utilizan para simplificar procesos de alta complejidad, en los que otros sistemas no pueden dar solución. La aplicación de este tipo de tecnología en los procesos de clasificación no solo puede mejorar la calidad y la eficiencia del procesamiento, sino también controlar el flujo de información haciendo que su visualización y análisis sea más sencillo. Este enfoque permite mejorar e incrementar la calidad en las operaciones del proceso obteniendo un

[1] Jonathan Iván Gutiérrez Gayosso alumno de la carrera de Ingeniería Mecatrónica de la Universidad Autónoma de Ciudad Juárez al159794@alumnos.uacj.mx
[2] El Dr. Luis Carlos Méndez González es Profesor Investigador del departamento de Ingeniería Industrial y Manufactura en la Universidad Autónoma de Ciudad Juárez luis.mendez@uacj.mx
[3] El Dr. Luis Alberto Rodríguez Picón es Profesor Investigador del departamento de Ingeniería Industrial y Manufactura en la Universidad Autónoma de Ciudad Juárez luis.picon@uacj.mx
[4] El Dr. Iván Juan Carlos Pérez Olguín es Profesor Investigador del departamento de Ingeniería Industrial y Manufactura en la Universidad Autónoma de Ciudad Juárez ivan.perez@uacj.mx

mejor desempeño en el tiempo y proporcionando una calidad superior.

Dentro del aprendizaje automático tenemos el campo del aprendizaje profundo que se basa en la utilización de redes neuronales artificiales inspiradas en la estructura y función del cerebro humano[4]. Estas redes neuronales se caracterizan por su capacidad de aprender representaciones automáticas de los datos de entrada. Esto se logra a través de la construcción de una jerarquía de capas de procesamiento, donde cada capa aprende una representación cada vez más abstracta de los datos recibidos de la capa anterior. La estructura de una red neuronal profunda es similar a una pila de capas, donde cada capa es responsable de aprender la representación de los datos de entrada del nivel anterior, de esta manera cada nivel se retroalimenta del nivel anterior. La primera capa es la capa de entrada, que recibe los datos brutos. A continuación, los datos pasan a través de varias capas ocultas, que van aprendiendo representaciones cada vez más abstractas de los datos. Finalmente, los datos llegan a la capa de salida, donde se producen las predicciones del algoritmo.
El aprendizaje profundo ha tenido una amplia gama de aplicaciones como: minería de datos, motores de búsqueda, aprendizaje multimedia, reconocimiento de voz, sistemas de recomendación, procesamiento del lenguaje natural y otros campos relacionados. también tiene grandes aplicaciones en el área de visión por computadora debido a que ayuda a analizar, procesar y clasificarla información con un gran porcentaje de efectividad[5].

Dentro del aprendizaje profundo se hace uso de las redes neuronales convolucionales (CNN por sus siglas en ingles). Estas redes neuronales artificiales están diseñadas especialmente para trabajar con imágenes y diferentes tipos de datos espaciales. Estas redes se basan en la idea de aplicar filtros a los datos de entrada para extraer características relevantes y generar una representación más robusta y compacta de los datos. En una CNN, la capa de entrada es un tensor de imagen, y las capas intermedias son capas de convolución que aplican filtros a los datos de entrada. Los filtros se deslizan sobre la imagen y realizan una operación matemática para detectar patrones específicos, como bordes o texturas. Estos patrones se agrupan en mapas de características, que luego son procesados por capas adicionales de la red para producir una representación compacta de los datos de entrada. Las CNNs son muy eficaces para realizar tareas de visión artificial, como la segmentación de imágenes, la detección de objetos y la recreación de imágenes. Además, debido a su capacidad para aprender características relevantes de los datos, las CNNs son ampliamente utilizadas en una variedad de aplicaciones, incluyendo la medicina, la robótica y la vigilancia por video.

En la actualidad existen diversos algoritmos de visión por computadora ya predefinidos en los que solo debemos ajustar ciertos parámetros y proveer al modelo con suficiente cantidad de datos. Entre estos algoritmos se encuentran: Faster R-CNN (Region-based Convolutional Neural Network), SSD (Single Shot multi-box Detector) y YOLO (You Only Look Once), estos algoritmos se especializan en la detección de objetos y son superiores a los algoritmos tradicionales en cuanto a precisión y velocidad de detección. Esta superioridad se debe a su estructura basada en aprendizaje profundo que hace uso de las redes neuronales convolucionales para lograr un desempeño excepcional.

El funcionamiento básico de SSD consiste en utilizar una red neuronal para realizar la clasificación de objetos en una imagen y la localización de estos objetos simultáneamente. La red neuronal está compuesta por varias capas conocidas como "multibox" las cuales generan diferentes tamaños y aspectos de cajas delimitadoras (bounding boxes) para cada objeto detectado. Cada caja delimitadora es una representación rectangular que envuelve a un objeto detectado y se asocia con una probabilidad que indica la confianza en que el objeto está realmente presente en la imagen. SSD utiliza una técnica llamada Non-Maximum Suppression (NMS) para filtrar las cajas delimitadoras superpuestas y seleccionar la caja con la mayor probabilidad para cada objeto detectado[6].

Uno de los aspectos más importantes de SSD es su desempeño a la hora de detectar objetos en diferentes escalas de tamaño en una sola imagen. Esto se logra mediante la utilización de múltiples tamaños de cajas delimitadoras generadas por diferentes capas de la red neuronal. Además, SSD también puede ser entrenado para detectar objetos en diferentes niveles de resolución, lo que le permite ser eficaz en la detección de objetos tanto pequeños como grandes en una sola imagen.

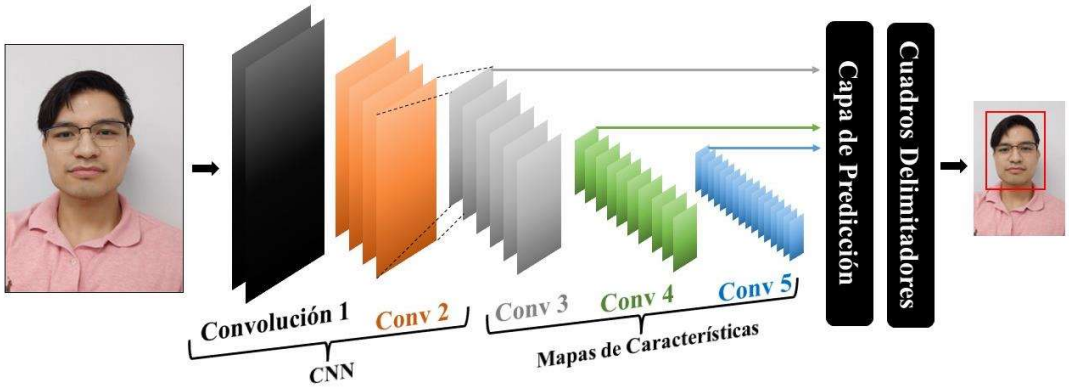

Figura 1. Arquitectura de SSD

Al igual que SSD, YOLO es un enfoque de "tiro único" para la detección de objetos, lo que significa que realiza la detección en una sola pasada a través de la red neuronal. En YOLO, se divide la imagen en una malla de celdas y se asigna a cada celda la responsabilidad de detectar objetos en su área. Cada celda genera una caja delimitadora para los objetos que detecta y asocia con ellos una probabilidad que indica la confianza en que el objeto está realmente presente en la imagen[7].

A diferencia de SSD, YOLO no utiliza capas específicas para detectar objetos de diferentes tamaños. En su lugar, YOLO utiliza una red neuronal de profundidad completa que se entrena para detectar objetos de cualquier tamaño. Este tipo de arquitectura hace que YOLO y SSD estén a la par en termino de capacidades, excepto en la detección de objetivos de gran tamaño debido a que la malla de celdas que utiliza puede resultar en la pérdida de detalles en la imagen, esto ocurre especialmente en objetos de tamaño grande.

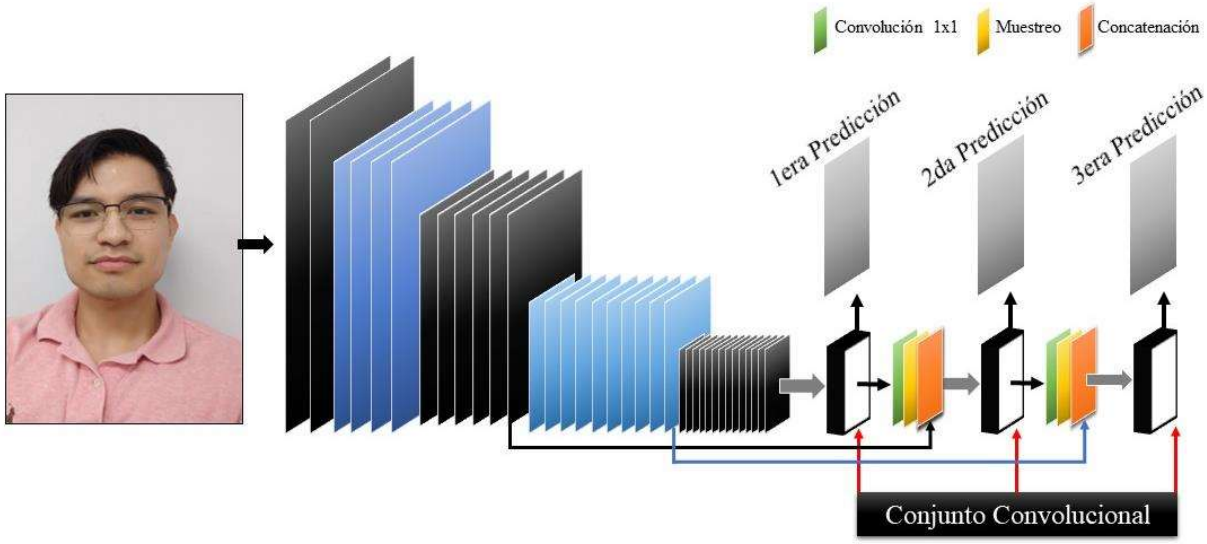

Figura 2. Arquitectura de YOLOv3

Faster R-CNN es una técnica de detección de objetos utilizada en imágenes, vídeos y grabaciones en tiempo real. A diferencia de YOLO y SSD, que son técnicas de "tiro único", Faster R-CNN es un enfoque de dos fases para la detección de objetos. La primera fase de Faster R-CNN es una red de región de propuestas (RPN), donde se generan candidatos para posibles regiones que contengan objetos. Estas regiones se utilizan como entrada para la segunda fase, que es una red neuronal de detección de objetos. La red neuronal en esta segunda fase analiza cada región propuesta para determinar si contiene un objeto y, en caso afirmativo, identifica la clase y localiza la caja delimitadora para el objeto[8]. Faster R-CNN es conocido por su alta precisión en la detección de objetos, especialmente para objetos de tamaño mediano y grande. Sin embargo, este enfoque también es más lento que YOLO y SSD debido a su fase adicional de propuestas de región.

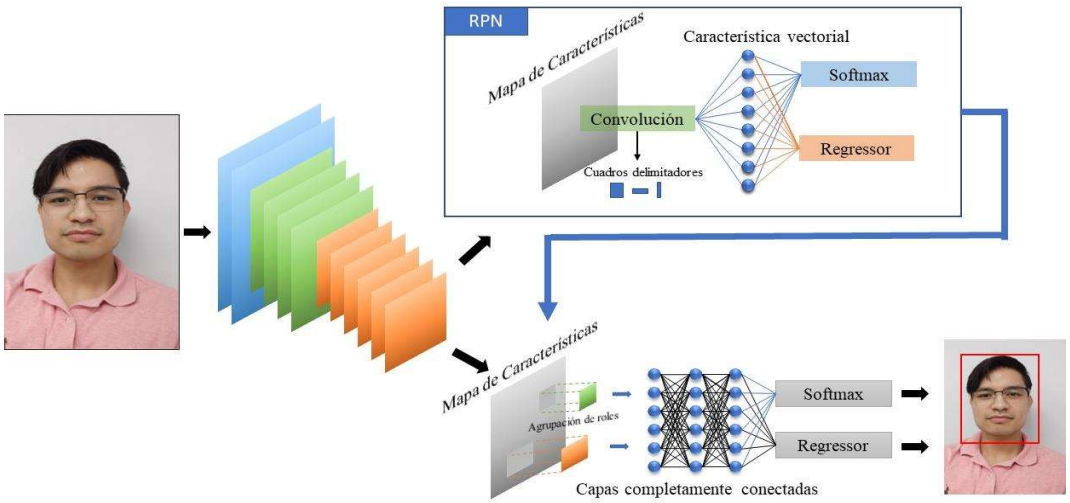

Figura 3. Arquitectura de Faster R-CNN

Para la ejecución de los 3 modelos se utilizó Google Colaboratory, también conocido como "Colab", es una plataforma gratuita de Jupyter Notebook alojada en la nube y mantenida por Google. Colab proporciona un entorno de desarrollo interactivo para la creación, ejecución y colaboración de proyectos de aprendizaje automático, análisis de datos y programación en Python. Los usuarios pueden cargar y compartir notebooks de Jupyter, que contienen código ejecutable, texto explicativo, visualizaciones y otros tipos de contenido multimedia. Además, Colab incluye acceso gratuito a GPUs y TPUs (unidades de procesamiento tensorial) para acelerar el entrenamiento de modelos de aprendizaje automático y otros tipos de tareas intensivas en cómputo. Colab también se integra con otros servicios de Google, como Google Drive y Google Cloud Storage, lo que permite a los usuarios almacenar y acceder a datos y modelos de manera conveniente.

Los usuarios pueden colaborar en tiempo real en un notebook y compartirlo con otros para facilitar la colaboración en proyectos. En general, Colab es una herramienta útil y poderosa para los científicos de datos, los ingenieros de aprendizaje automático y los desarrolladores de software que desean trabajar en proyectos de manera colaborativa y sin necesidad de configurar un entorno de desarrollo en sus propias máquinas.

Como base para SSD y Faster R-CNN se utilizó la red convolucional ResNet50. A pesar de que ResNet50 es una tecnología diferente a SSD y Faster R-CNN, ResNet50 también se utiliza para tareas de visión por computadora por lo que se pueden combinar para mejorar el rendimiento de los modelos de detección de objetos. En esta combinación ResNet50 se utiliza como base para extraer las características de las imágenes en diferentes escalas y niveles de abstracción en las tareas de clasificación mientras que, SSD y Faster R-CNN se enfocan en la detección del objeto delimitando el área de interés y clasificando el objeto detectado. La combinación de ResNet50 y estos modelos se ha utilizado en muchas aplicaciones de visión por computadora, incluyendo la detección de objetos en imágenes de satélite, la detección de objetos en imágenes médicas y la detección de objetos en videos[9]. La combinación de ambas tecnologías permite construir modelos de detección de objetos de alto rendimiento que son precisos y rápidos.

En el caso de YOLOv3 este modelo posee su propia red neuronal convolucional, llamada Darknet53. La arquitectura de Darknet53 se basa en el concepto de conexiones residuales, similar a la arquitectura de ResNet. Esto permite que la información fluya directamente a través de las capas en lugar de tener que pasar por múltiples capas, lo que permite construir redes más profundas y efectivas. Darknet53 se compone de 53 capas convolucionales y utiliza bloques residuales para construir una arquitectura profunda. En particular, utiliza bloques de convolución, normalización por lotes, activación ReLU y capas de agrupación máxima para extraer características de las imágenes. Además, utiliza capas de conexión completa para generar predicciones finales[7].

Para el entrenamiento de los 3 modelos se utilizó el mismo conjunto de datos COCO 2017. Una base de datos ampliamente utilizada para tareas de reconocimiento y detección de objetos en imágenes. "COCO" es un acrónimo de "Common Objects in Context" (Objetos Comunes en Contexto), que hace referencia a la naturaleza de la base de datos, que se enfoca en imágenes que muestran objetos comunes en situaciones cotidianas.

La base de datos COCO 2017 contiene más de 330,000 imágenes etiquetadas con más de 2.5 millones de instancias de objetos, lo que la convierte en uno de los conjuntos de datos más amplio e integro de su tipo. Además,

cada imagen en el conjunto de datos viene con múltiples anotaciones que incluyen la clase del objeto, la posición del objeto en la imagen y una máscara de segmentación detallada para cada objeto. Además de las anotaciones de objetos, COCO 2017 también incluye anotaciones para otras tareas, como detección y segmentación de personas, detección y segmentación de rostros, y detección de puntos clave humanos. El conjunto de datos COCO 2017 se utiliza ampliamente para entrenar y evaluar algoritmos de reconocimiento y detección de objetos en imágenes, especialmente en el ámbito de la visión por computadora y el aprendizaje profundo. Debido a su tamaño y calidad, se considera una de las bases de datos más importantes en esta área.

Para llevar a cabo la comparación de los 3 modelos se realizaron 50 detecciones en diferentes imágenes con distintos tipos de objetivos como vehículos, personas, animales, alimentos y diferentes objetos misceláneos. En las detecciones se tomaron en cuenta los valores de certeza que calcula el algoritmo al momento de clasificar el objeto y el tiempo de ejecución de cada modelo en cada una de las detecciones. También se tomaron en cuenta las lecturas en donde el modelo detecto un objeto en donde no se encontraba ningún objetivo (falsa detección), de igual manera los casos en donde el modelo no detecto un objeto en donde si se encontraba un objetivo de detección (detección eludida), esas lecturas se capturaron con valor de 0. En base a los valores capturados se sacó el promedio del tiempo de ejecución y del porcentaje de certeza de cada modelo. Con estos resultados se obtuvo que algoritmo es el más rápido en tiempo de ejecución y el de mayor grado de certeza.

Resultados

	SSD	YOLOv3	Faster R-CNN
Tiempo de ejecución	3.31s	5.14s	44.01s
Porcentaje de certeza	77.37%	74.87%	82.75%
Falsa detección	3	3	5
Detección eludida	5	4	1

Cuadro 1. Resultados de los 3 modelos.

Faster R-CNN es el modelo más preciso llegando a reconocer objetivos que los otros modelos pasan por alto, sobretodo cuando se trata de objetos muy pequeños o que solo se logran apreciar parcialmente. Pero su tiempo de ejecución es muy superior al de SSD y YoloV3 llegando a tardarse hasta 10 veces más que los otros modelos. Tambien tiene problemas con los falsos positivos llegando a detectar objetos donde no los hay, este problema podria ser debido a que Faster R-CNN tiende a sobreprocesar algunas imágenes.

Por otro lado, SSD logra obtener mejor desempeño en cuanto a precisión y velocidad frente a YoloV3, lo que lo vuelve el algoritmo mas equilibrado. Ambos modelos tienen problemas con las detecciones eludidas en donde los modelos no fueron capaces de detectar un objeto donde claramente habia uno. Tambien, tienen problemas con los falsos positivos aunque en menor medida que Faster R-CNN.

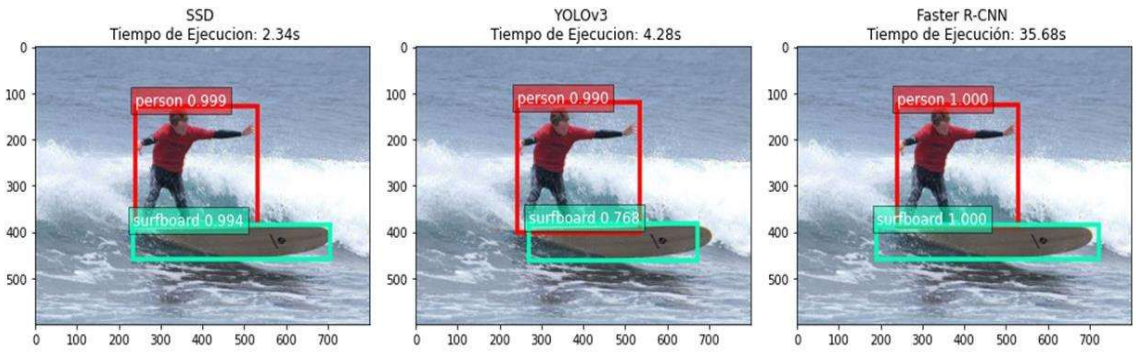

Figura 4. Resultado de los 3 modelos para una misma imagen.

Conclusiones

Se puede concluir por los resultados que Faster R-CNN es un modelo más apropiado para ser utilizado en sistemas de clasificación, donde la precisión es la prioridad. Mientras que los modelos YoloV3 y SSD son más adecuados para sistemas de detección en tiempo real, donde el tiempo de ejecución es un factor crítico en los resultados del sistema. En cualquier caso, la elección del modelo adecuado dependerá de los requisitos específicos de cada aplicación. Además, el desempeño también dependerá en gran medida de los datos que suministremos para el

entrenamiento del sistema. Dando a nuestro sistema mayor variedad de datos obtendrá una mayor flexibilidad mientras que un suministro de datos muy específico dotará al sistema con una precisión excepcional en circunstancias controladas.

Referencias

[1] J. Lemley, S. Bazrafkan and P. Corcoran, “Learning data augmentation for consumer devices and services,” 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2018.

[2] P. Constante, A. Gordon, O. Chang, E. Pruna, F. Acuna and I. Escobar, "Artificial Vision Techniques to Optimize Strawberry's Industrial Classification," in IEEE Latin America Transactions, vol. 14, no. 6, 2016.

[3] H. H. Álvarez-Valera, E. Bolivar-Vilca, C. Cervantes-Jilaja, E. E. Cuadros-Zegarra, D. Barrios-Aranibar and R. Patiño-Escarcina, "Automation of Chestnuts Selection Process Using Computer Vision in Real Time," 2014 33rd International Conference of the Chilean Computer Science Society (SCCC), Talca, Chile, 2014.

[4] F. Ertam and G. Aydın, "Data classification with deep learning using Tensorflow," 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey, 2017.

[5] H. Li, J. Li, X. Guan, B. Liang, Y. Lai and X. Luo, "Research on Overfitting of Deep Learning," 2019 15th International Conference on Computational Intelligence and Security (CIS), Macao, China, 2019.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot MultiBox detector,” in Computer Vision – ECCV 2016. Springer International Publishing, 2016.

[7] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” 2018. [Online]. Available: https://arxiv.org/abs/1804.02767

[8] S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, 2017.

[9] D. Kumar, X. Zhang, H. Su and S. Wei, "Accurate Object Detection Based on Faster R-CNN in Remote Sensing Imagery," 2019 6th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR), Xiamen, China, 2019.