

ESTUDIO Y COMPARATIVA DE ALGORITMOS DE DETECCIÓN DE OBJETOS CON REDES NEURONALES ARTIFICIALES CONVOLUCIONALES PARA LA DETECCIÓN DE ENFERMEDADES EN HOJAS.

STUDY AND COMPARISON OF OBJECTS DETECTION ALGORITHMS USING CONVOLUTIONAL NEURAL NETWORKS FOR PLANT DISEASES DETECTION IN LEAVES.

Ángel Omar González Frayre, Carlos Alejandro Guerrero Méndez, Carlos Alberto Olvera Olvera*, Francisco Eneldo López Monteagudo, Daniela López Betancur, Valeria Maeda Gutiérrez, Santiago Villagrana Barraza. Unidad Académica de Ingeniería Eléctrica, Universidad Autónoma de Zacatecas, Jardín Juárez 147, Centro, Zacatecas 98000, Zac. México. omar11093@hotmail.com, colvera@uaz.edu.mx*

Resumen

Recientemente el Machine Learning y la visión computacional han originado interés y han encontrado nuevas aplicaciones en la ingeniería. En la agricultura, sistemas “inteligentes” se han convertido en herramientas importantes para la detección de anomalías que decrecen la calidad y cantidad en la cosecha de productos agrícolas. En esta investigación desarrollamos una comparativa de los principales algoritmos de detección de objetos usando Redes Neuronales Convolucionales (RNC) implementadas en Deep Learning. Se analizaron los resultados en base a la superposición y tiempo de procesamiento obtenidos con los algoritmos de detección de objetos R-CNN, Fast R-CNN, y Faster R-CNN. Se implementaron las arquitecturas de RNC de AlexNet, GoogleNet, ResNet50, ResNet101, SqueezeNet e InceptionV3, para generar Transfer Learning en los detectores y clasificadores de imágenes. Las arquitecturas fueron entrenadas con PlantVillage - Dataset, conformado por más 40,000 imágenes de hojas de 9 especies de plantas y 24 enfermedades.

Palabras clave: Inteligencia Artificial aplicada a la ingeniería, Detección de enfermedades de plantas, Redes neuronales convolucionales, Deep Learning.

Abstract

Recently Machine Learning and computational vision have generated interest and have found new applications in engineering. In agriculture, "smart" systems have become important tools in the detection of anomalies that decrease the quality and quantity in the harvest of agricultural products. In this research, we developed a comparison of the main object detection algorithms using Convolutional Neural Networks (CNN) implemented in Deep Learning. The results were analyzed based on the accuracy and processing time obtained with the object detection algorithms R-CNN, Fast R-CNN, and Faster R-CNN. The CCN topologies of AlexNet, GoogleNet, ResNet50, ResNet101, SqueezeNet and InceptionV3 were implemented to generate Transfer Learning in image detectors and classifiers. The topologies were trained with the PlantVillage - Dataset which is made up of more than 40,000 leaf images of 9 plant species and 24 diseases.

Keywords: Artificial intelligence applied to engineering, Detection of plant diseases, Deep Learning.

1. Introducción

Las enfermedades, plagas y otros agentes indeseados presentes en cultivos agrícolas pueden significar una gran reducción en la producción de los productos del campo (Strange & Scott, 2005). Las afectaciones de estas

entidades perjudiciales para el cultivo se ven reflejadas directamente en la disminución de la calidad y cantidad de la cosecha de un cultivo. Para contrarrestar, regular y manejar las afectaciones de los organismos biológicos y enfermedades se ha ideado el término “control de plagas” (Rangarajan et al, 2018). Comúnmente, la detección de plagas y enfermedades en cultivos suele diagnosticarse mediante una exploración visual en base a la apariencia,

morfológica y otros aspectos de la hoja. Se recomienda que dichos estudios visuales solo sean realizados y diagnosticados por biólogos expertos que cuenten con una elevada experiencia, ya que un diagnóstico erróneo puede significar pérdidas irreparables en el cultivo. Se debe de mencionar que la realización de estudios de control de plagas y enfermedades suelen ser costosos y requieren de la permanencia del biólogo experto para ser diagnosticado de la manera más rápida posible y evitar la propagación y el contagio de alguna enfermedad (Nutter et al, 2006).

Recientemente, la Inteligencia Artificial (IA) ha encontrado una gran cantidad de aplicaciones en todos los ámbitos de la vida cotidiana, lo que ha propiciado de la aparición de los términos Machine Learning (ML) y Deep Learning (DL), que, en simples palabras, originan que las máquinas puedan “aprender” patrones usando una gran cantidad de datos para posteriormente realizar acciones. La unión del DL y técnicas de visión artificial ha originado algoritmos inteligentes que realizan detección y clasificación de patrones o imágenes con un desempeño más acertado que el que el humano promedio podría realizar (Barbedo, 2019).

Debido a lo anterior han surgido técnicas computacionales para la detección automática Mohanty (Mohanty et al, 2016), quienes desarrollaron un sistema de diagnóstico de enfermedades de hojas de plantas asistido por teléfonos inteligentes, empleando dos arquitecturas de Redes Neuronales Convolucionales (RNC), AlexNet (Krizhevsky et al, 2012) y GoogleNet (Szegedy et al, 2015). Su modelo propuesto se entrenó y alcanzó una exactitud de 99.35% en la detección de enfermedades de hojas. Para los algoritmos de detección (Quan et al, 2019) utilizaron algoritmos de detección faster R-CNN en combinación con redes neuronales convolucionales (RNC) para detectar plantas de maíz en crecimiento en sus fases de inicio y diferenciarlas de las malas hierbas en 3 diferentes condiciones climáticas teniendo resultados de detección en promedio de 97.71%. En otra investigación Yu, (Yu et al, 2019) utilizó las técnicas de Deep learning donde ocupó el detector faster en combinación con la red neuronal ResNet-50 para detectar frutas para posteriormente implementarlo en un robot cosechador de fresas, los resultados de detección obtenidos en esta investigación fueron en la precisión de 95.78% y de superposición de 89.85%. Fuentes (Fuentes et al, 2017) comparó tres diferentes tipos de familias de detectores en combinación con varias arquitecturas, teniendo mejor desempeño en los resultados con faster R-CNN en combinación con la arquitectura VGG-16.

En este trabajo de investigación realizamos una comparativa de los algoritmos de detección de imágenes

más usados para la detección y clasificación de objetos. Nuestro trabajo busca estimular cuál de estos algoritmos prevé un mejor desempeño en la detección de hojas enfermas. Se espera que los resultados obtenidos sirvan para determinar cuál de los algoritmos es el más conveniente para ser implementado en la creación de un sistema “inteligente” de detección de enfermedades en hojas.

2. Materiales y métodos

Un par de las aplicaciones desarrolladas con el DL y la visión artificial son: la clasificación y detección de objetos. Básicamente, el proceso de detección busca en una imagen aquella región que cumpla con las características con las que se entrenó el algoritmo de detección, una vez localizado el objeto se procede a la generación de una caja delimitadora que encierra al objeto, con lo cual obtenemos las coordenadas espaciales del objeto. Por otro lado, la clasificación se encarga en categorizar el objeto de mayor dimensión en una imagen (Girshick et al, 2014).

La comparación de los algoritmos de detección se hará en base a los detectores de R-CNN, Fast R-CNN, y Faster R-CNN, que, si bien existen diferencias del funcionamiento de estos, su funcionamiento fue implementado según las configuraciones señaladas en la documentación del software MatLab R2018b (Matworks, 2018).

Generalmente para el entrenamiento de un detector de imágenes se sugiere realizar los siguientes pasos:

- Adquisición de imágenes,
- Entrenamiento del modelo de detección
- Evaluación del modelo.

2.1 Adquisición de imágenes.

Los sistemas inteligentes de detección y clasificación de imágenes requieren de una cantidad de información para ser entrenados. Por lo general los sistemas de ML y DL mejoran su desempeño al ser entrenados con una elevada cantidad de datos.

En este proyecto recurrimos a la implementación de la base de datos PlantVillage (Hughes et al, 2015). Este “dataset” está conformado por 54,323 imágenes de 14 cultivos y divididas entre 38 clases de hojas de plantas sanas y que presentan algún tipo de enfermedad. Para este estudio se utilizaron 40,000 imágenes de 9 tipos de vegetales, Manzana (3,171 imágenes), Cereza (1,906 imágenes), Maíz (3,852 imágenes), Uva (4,062 imágenes), Durazno (2,657 imágenes), Pimiento (2,475 imágenes), Papa (2,152

imágenes), Fresa (1,565 imágenes), Tomate (18,160 imágenes).

Para el entrenamiento y evaluación de un sistema de DL es necesario dividir el dataset en un conjunto de entrenamiento y uno de evaluación. Los datasets generados en esta investigación tendrán la peculiaridad de tener imágenes con diferentes tamaños, otorgando mayor robustez a los detectores. Esto se consigue segmentando las regiones pertenecientes a las hojas en el PlantVillage. Después, cada región segmentada fue insertada en el centro de una imagen vacía con un valor de fondo de 128 en los canales RGB y de tamaño aleatorio. Para creación del conjunto de datos de entrenamiento se seleccionaron 50 imágenes aleatoria de cada una de las 33 clases de hojas, dando un total de 1650 imágenes en el dataset de entrenamiento. Por otro lado, el dataset de validación fue elaborado 5 imágenes de cada tipo del PlantVillage, dando un total de 165 imágenes, que equivale al 10% del número de imágenes usadas para el entrenamiento.

2.2 Entrenamiento del modelo de detección.

Los algoritmos de detección de objetos implementados en esta investigación deben de ser configurados modificando y agregando determinadas capas de las arquitecturas de la RNC que se implementa para la detección. Para el caso de la configuración de la RNC en el detector de objetos de R-CNN se eliminaron las últimas 3 capas de la arquitectura y se agrega una fullyConnectedLayer, softmaxLayer y classificationLayer. Las capas creadas se conectaron a una capa de salida en donde se obtienen los resultados (coordenadas de la región de interés). Para el caso de las RCN en los detectores Fast-RCNN, se realizó lo mismo que en el R-CNN, pero además se agregó una capa boxRegressionLayers y una de ROIpoolinglayer, que se conectaron a una capa roiInputLayer. Además, en las RCN se eligió una capa que se usará como featureExtractiuonLayer. Para el caso de las RNC con el detector de Faster R-CNN, se realizan los mismos pasos que en el R-CNN y Fast R-CNN, aquí además se agregando una RegionalProposalNetwork (RPN). Las capas modificadas y agregadas se realizaron según las sugerencias y especificaciones encontradas en la documentación de MatLab R2018b (Mathworks, 2018).

Técnicamente los detectores de objetos realizan la detección y clasificación de la región detectada, aunque esto implica un gran costo computacional durante el proceso de entrenamiento. Se decide entrenar los detectores de objetos usando una pequeña porción del dataset de entrenamiento para que sean capaz de detectar la hoja (sin importar su clase) en la imagen, después, la hoja detectada será clasificada utilizando una RNC que fue

entrenada previamente para este proceso (Maeda et al, 2018).

El proceso de entrenamiento es fundamental para la obtención de un detector de imágenes que funcione adecuadamente. Para esta comparativa se utilizarán las mismas opciones de entrenamiento de los algoritmos de clasificación. Los parámetros de entrenamiento se configuran antes del entrenamiento del modelo y no forman parte de los modelos como tal, son ajustables y se eligen para entrenar un modelo y que rigen el propio proceso de entrenamiento, el rendimiento del modelo depende en gran medida de los valores de los parámetros seleccionados.

En este caso en cada arquitectura se utilizaron los mismos hiperparámetros y se muestran en la Tabla 1.

- La opción de entrenamiento: descenso de gradiente estocástico con impulso “sgdm”.
- Taza de aprendizaje de 0.1 cada 10 épocas.
- El número de épocas se estableció en 20.
- El tamaño de lotes (MiniBatchSize) se estableció en 1 para los detectores Fast R-CNN y Faster R-CNN, y 4 para R-CNN.

Tabla 1. Los parámetros de entrenamiento.

Parámetros de las opciones de entrenamiento.	
Opción	Parámetro seleccionado
trainingOptions	sgdm
MiniBatchSize	1 - 4
LearnRateDropFactor	0.1
LearnRateDropPeriod	10
MaxEpochs	20

Para la realización de la clasificación con una RNC es importante conocer la profundidad y los parámetros de entrada de esta. Las capas de entrada de la RNC definirán el tamaño de la imagen de entrada a ser clasificada. Los parámetros establecidos de las RNC utilizadas para la clasificación de imágenes pueden consultarse en la documentación de MatLab R2018b (Mathworks, 2018).

El PC usado para el entrenamiento cuenta con una GPU que es una tarjeta de procesamiento gráfico. La implementación de la GPU disminuirá considerablemente el tiempo de entrenamiento de la RNC. Las características del computador utilizado son enlistadas en la Tabla 2.

Tabla 2. Características de la estación de trabajo.

Características del equipo	
Hardware	Características
Memoria RAM	16 GB
Procesador	Intel Core i7-7700 CPU @ 3.60 GHz
Unidad de Procesamiento Gráfico (GPU)	GeForce GTX 1070 X 8 Gb
Sistema operativo	Windows 10, 64 bits

2.3 Métricas de evaluación.

Según las métricas obtenidas en el proceso de evaluación es que podemos definir cuál es el mejor detector disponible para la detección de hojas. Para la comparación del desempeño de los detectores implementaremos la métrica de superposición, también llamada Intersección sobre unión, que se refiere al empalme entre las coordenadas del objeto detectado y las coordenadas del objeto en la imagen Ecuación (1).

$$\text{Superposición} = \frac{\text{Intersección}}{\text{Objeto} + \text{Detección}} \quad (1)$$

donde Interferencia se refiere al área que comparten el área del objeto en la imagen (Objeto) y el área de la detección realizada por el algoritmo (Detección). Objeto + Detección significan el área tanto del Objeto y de la Detección.

La Fig. 1 muestra el cálculo de Superposición o Intersección sobre la Unión.

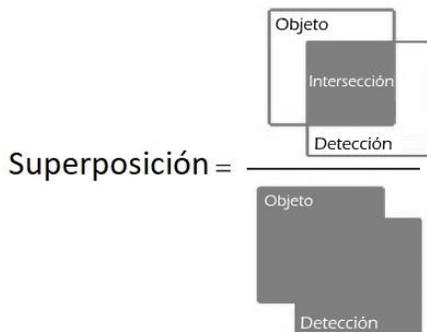


Fig. 1. Métrica de detección de objetos.

A continuación, se muestra en la Fig. 2. cómo es que dependiendo del valor de Intersección Sobre la Unión es que se puede evaluar la detección de un algoritmo.

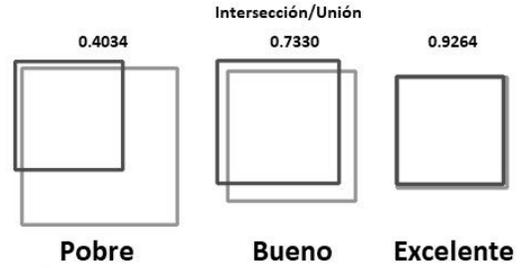


Fig. 2. Funcionamiento de la intersección sobre la unión.

Para establecer valor de superposición se utilizará el valor promedio en todo el dataset de validación.

3. Resultados

Los resultados obtenidos en esta investigación fueron realizados usando las detecciones arrojadas por los algoritmos de R-CNN, Fast R-CNN y Faster R-CNN en el dataset de validación.

Los algoritmos de detección de objetos encuentran y clasifican la región de interés encontrada por una RNC. Sin embargo, en la realización de este proyecto se recurrió a generar dichos procesos por separados, esto con el fin de aminorar la carga computacional. Se generó un detector de objetos que encontrará el objeto de interés y posteriormente se utilizó un clasificador (ya entrenado) de imágenes de hojas de plantas el cual nos señalará el nombre del objeto que se detectó.

Tabla 3. Resultados de clasificación.

Resultados del proceso de clasificación		
Arquitectura de CNN	Porcentaje de exactitud	Tiempo de entrenamiento
AlexNet	98.90%	1,360 min 13 seg
GoogleNet	98.85%	2,639 min 49 seg
InceptionV3	96.22%	7,143 min 5 seg
SqueezeNet	97.37%	1,217 min 34 seg
ResNet 50	97.56%	4,878 min 6 seg
ResNet 101	98.07%	7,672 min 15 seg

A continuación, se explicará el funcionamiento del algoritmo para calcular las métricas de comparación:

- Se inserta una imagen con la hoja enferma del cultivo de la que se requiere obtener su detección. Para este ejemplo

se usó la imagen corresponde a una hoja sana a de tomate como salida (ver Fig. 3), el algoritmo resaltará el objeto detectado en la imagen de entrada.

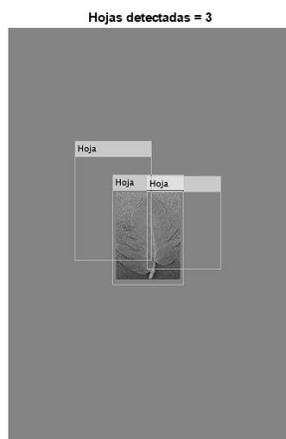


Fig. 3. Ejemplo de hoja detectada.

Posiblemente, el algoritmo detecte dos o más objetos en donde realmente existe uno solo. Se procederá a obtener las calificaciones (scores) con el parecido en la detección de objetos arrojada por la RNC. Dependiendo de las calificaciones obtenidas se seleccionará la caja de las coordenadas con la calificación más alta, con lo anterior se garantiza la obtención de la caja delimitadora que contenga el objeto detectado más parecido a los utilizados en el entrenamiento.

Una vez obtenido la caja delimitadora con mayor calificación se procederá a clasificar el objeto detectado por el algoritmo de detección. Se deberán modificar las dimensiones de la caja delimitadora para que pueda ser la imagen de entrenada de la RNC en la que se realizará la clasificación del objeto, es decir, las dimensiones de la imagen a clasificar deberán coincidir con la capa de entrada de la RNC.

Para la evaluación de cuál algoritmo de detección es el mejor para realizar la detección de hojas enfermas se procede al cálculo de la métrica de Superposición. Utilizamos las cajas del objeto, detección vistas en la Fig. 4. Además, en el resultado final podemos ver la caja con la clasificación de objeto.



Fig. 4. Cuadro de detección, cuadro del objeto y cuadro con el resultado de la clasificación.

Con la elaboración de esta investigación buscamos determinar cuál algoritmo de detección de objetos arroja mejores resultados en la detección de hojas con alguna enfermedad. Los resultados obtenidos con esta investigación se enlistan en la Tabla 4.

Tabla 4. Resultados obtenidos de la detección de hojas.

Resultados de detección	
Detector / CNN	Superposición
R-CNN / AlexNet	0.5419
R-CNN / GoogleNet	0.7197
R-CNN / InceptionV3	0.6775
R-CNN / SqueezeNet	0.5949
R-CNN / ResNet 50	0.8122
R-CNN / ResNet 101	0.7058
Fast R-CNN / AlexNet	0.8006
Fast R-CNN / GoogleNet	0.8626
Fast R-CNN / InceptionV3	0.7119
Fast R-CNN / SqueezeNet	0.6753
Fast R-CNN / ResNet 50	0.7388
Fast R-CNN / ResNet 101	0.7276
Faster R-CNN / AlexNet	0.8508
Faster R-CNN / GoogleNet	0.9235
Faster R-CNN / InceptionV3	0.6985
Faster R-CNN / SqueezeNet	0.6814
Faster R-CNN / ResNet 50	0.8339
Faster R-CNN / ResNet 101	----

El detector Faster R-CNN con la arquitectura de ResNet-101 no fue entrenado debido a que el computador utilizado no cuenta con los requerimientos mínimos para realizar el proceso de entrenamiento.

4. Conclusiones

El Deep Learning (DL) ha logrado grandes avances en el campo de la investigación en detección y clasificación de enfermedades en plantas y cultivo. En esta investigación se han comparado los tres principales algoritmos de detección de imágenes (R-CNN, Fast R-CNN y Faster R-CNN) con el fin de determinar cuál algoritmo es más efectivo en su implementación de un autómata que ayude en la detección de enfermedades en plantas. Dado los resultados arrojados podemos definir que el detector Faster R-CNN con GoogleNet es el más conveniente al ser utilizado en un autómata detector de hojas sanas y enfermas. Además, en esta investigación comprobamos que es posible utilizar RNCs de clasificación de imágenes en sistemas de detección, con lo que logramos reducir el costo computacional requerido para el entrenamiento de un algoritmo de detección.

Referencias

Barbedo, J. (2019). Plant disease identification from individual lesions and spots using deep learning. *Biosystems Engineering* 180 96-107

Fuentes, A., Yoon, S., Kim S., & Park D. (2017). A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. *Sensors* 2, 17(9)

Girshick, R., Donahue J., Darrell T. & Malik, J., (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524v5*.

Hughes, D., & Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. En *Advances in neural information processing systems* (págs. 1097-1105).

Maeda, V., Guerrero C., Olvera C., Araiza M., Espinoza G., & Bordón, R., (2018). Redes neuronales convolucionales para la detección y clasificación de enfermedades de plantas basadas en imágenes digitales. *Revista Biológico Agropecuaria Tuxpan (Especial Congreso)* (6) (págs. 275-282)

Matworks. (2018). Train a Fast R-CNN Deep Learning Object Detector (r2018b) Retrieved July 6, 2019 from <https://la.mathworks.com/help/vision/ref/trainfastrcnno bjectdetector.html>

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for imagebased plant disease detection. *Frontiers in plant science.*, 7, 1419.

Nutter, F.W., Jr.; Esker, P.D. & Coelho, R. (2006). Disease assessment concepts and the advancements made in improving the accuracy and precision of plant disease data. *Eur. J. Plant Pathol.* 115, 95–113.

Quan, L., Feng, H., Lv, Y., Wang, Q., Zhang, C., Liu, J., & Yuan, Z. (2019). Maize seedling detection under different growth stages and complex field environments based on an improved Faster R-CNN. *Biosystems Engineering* 184 1-23

Rangarajan, A., Purushothaman, R, Ramesh, A., (2018) Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Computer Science* 1331040–1047

Strange, R. N., & Scott, P. R. (2005). Plant disease: a threat to global food security. *Phytopathology* 43, 83–116.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., . . . Rabinovich, A. (2015). Going deeper with convolutions. En *Proceedings of the IEEE conference on computer vision and pattern recognition* (págs. 1-9).

Yu, Y., Zhang, K., Yang, L., & Zhang, D. (2019). Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. *Computers and Electronics in Agriculture* 163 104846