

Segmentación semántica en cultivos de jitomate usando ResNet50

Juan Pablo Guerra Ibarra¹, Francisco Javier Cuevas de la Rosa¹,
Oziel Arellano Arzola²

¹ Centro de Investigaciones en Óptica A.C.,
México

² Instituto Tecnológico de Estudios Superiores de Zamora,
México

{juangi,fjcuevas}@cio.mx, oziel.aa@zamora.tecnm.mx

Resumen. La aplicación de algoritmos de inteligencia artificial poco a poco van permeando en diferentes ámbitos de la vida cotidiana. Sirva de ejemplo, el uso de algoritmos de aprendizaje profundo han ido aumentando su presencia en diversos aspectos de las actividades de la sociedad en su día a día. Del mismo modo en las últimas dos décadas, la agricultura de precisión ha incrementando su impacto en los diferentes procesos de la labranza de la tierra, principalmente, en busca de aumentar su productividad mientras reduce su impacto al medio ambiente. La aplicación de algoritmos de aprendizaje profundo en la agricultura, genera un área de estudio para el desarrollo de diferentes tecnologías, las cuales apoyan una actividad primordial para mantener y preferentemente aumentar la producción de alimentos, lo que aseguran la subsistencia presente y futura de la sociedad moderna. En el presente trabajo se utiliza una red neuronal convolucional, en específico el modelo *ResNet50* para realizar la tarea de una segmentación semántica en imágenes de cultivos de jitomate en invernaderos, el objetivo es separar las hojas y frutos que estén presentes en la imagen. La segmentación de imágenes es la etapa inicial para el desarrollo de otros sistemas computacionales al eliminar el ruido presente y separar los elementos de interés, lo cual es básico para el desarrollo de mejores tecnologías aplicadas a los procesos agrícolas.

Palabras clave: Aprendizaje profundo, redes neuronales convolucionales, segmentación semántica, agricultura de precisión.

Semantic Segmentation in Tomato Crops Using ResNet50

Abstract. The application of artificial intelligence algorithms is gradually permeating various areas of everyday life. A clear example is the growing presence of deep learning algorithms in multiple aspects of

society's daily activities. Similarly, over the last two decades, precision agriculture has increased its impact on different farming processes, mainly aiming to enhance productivity while reducing its environmental impact. The use of deep learning algorithms in agriculture creates a research area for the development of different technologies that support a fundamental activity for maintaining—and ideally increasing—food production, ensuring the present and future subsistence of modern society. In this work, a convolutional neural network, specifically the ResNet50 model, is employed to perform semantic segmentation tasks on images of tomato crops in greenhouses. The goal is to separate the leaves and fruits present in the image. Image segmentation is the initial stage for developing other computational systems, as it eliminates noise and separates the elements of interest, which is essential for building better technologies applied to agricultural processes.

Keywords: Deep learning, convolutional neural networks, semantic segmentation, precision agriculture.

1. Introducción

La agricultura de precisión engloba un conjunto de tecnologías, las cuales combinan sensores, estadística, algoritmos clásicos, algoritmos inteligentes entre otros, con el objetivo de optimizar la producción de alimentos, tanto en cantidad como en calidad, esto se logra dando seguimiento al crecimiento de los cultivos y reduciendo costos [11,30,28]. Para lograr incrementar la producción de alimentos provenientes de los campos de cultivos, es importante realizar de manera oportuna y eficiente la detección de las necesidades que pudieran tener las plantas, algunas éstas pueden ser referentes a proceso de fertilización, riego é iluminación, por mencionar algunas.

La detección oportuna de las necesidades nutrimentales é hídricas es de gran relevancia en la agricultura protegida, ya que el riego y nutrición dependen de la eficiencia para determinar dichas necesidades [31,40,36]. Buscando apoyar la detección eficiente de necesidades en diferentes tipos de cultivos, se han realizado diferentes trabajos con técnicas de segmentación basadas en umbrales en formato de color *RGB* o en algún otro sistema de representación del mismo [8,42,32,7,15,37,29]. Los procedimientos de segmentación mencionados en los artículos [8,42,32,7,15,37,29] han reportado resultados exitosos en sus experimentos. Los trabajos citados tienen en común, el que las imágenes con las que se realizan los procesos de segmentado fueron tomadas en condiciones controladas; haciendo énfasis en controlar el fondo que se aprecia en las imágenes, con la variable de fondo controlada, se facilita en gran medida la determinación de los umbrales adecuados para llevar a cabo la separación o segmentación de los píxeles asignándolos a una clase particular.

En años recientes la rama de la inteligencia artificial llamada aprendizaje máquina, ha sido probada en diferentes espacios de la vida moderna, una de las actividades donde más impacto puede tener es en la agricultura de precisión

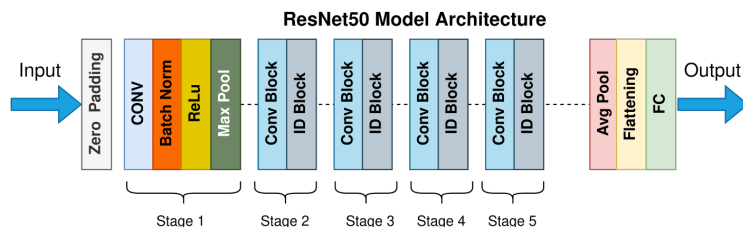


Fig. 1. Estructura de la RNC *ResNet50*.

[4,26,10,23], los algoritmos de aprendizaje profundo son los que mayor relevancia han tenido.

Los algoritmos de aprendizaje profundo han surgido como una alternativa para procesar grandes cantidades de información heterogénea, lo que facilita la tarea de realizar predicciones y clasificaciones complejas en ambientes agrícolas; actualmente existe una amplia diversidad o gama de sensores que generan grandes volúmenes de información, como lo son imágenes provenientes de diferentes tipos de dispositivos de captura, desde cámaras de teléfonos celulares hasta drones especializados en agricultura de precisión [28,35,16,6,25]. En la segmentación de imágenes adquiridas en ambientes agrícolas no controlados, con la finalidad de separar los elementos de estudio de interés del resto de la imagen, es una de las áreas donde se pueden explotar las ventajas de los algoritmos de aprendizaje profundo.

La segmentación semántica es la asignación de una clase o etiqueta a cada píxel que forman una imagen [12,22]. Llevar a cabo el proceso de etiquetado o marcado de los píxeles de una imagen mediante aprendizaje profundo ha tenido un crecimiento significativo en los años recientes [35]. Técnicas o herramientas de aprendizaje profundo que se ha empleado con éxito, son las redes neuronales convolucionales *RNC* [5,45,44], el éxito de este tipo de arquitectura se funda en su invariancia ante traslaciones y cambio de escalas en las entradas proporcionadas a los modelos de *RNC*, lo cual es excelente en problemas de alto nivel de ruido.

En la presente investigación se explora el uso de una *RNC*, en particular la *ResNet50* [13] (ver Figura 1 tomada de [2]), para desarrollar un segmentador semántico, el cual separe los píxeles que conforman las hojas y frutos de cultivo de jitomate en imágenes tomadas en invernaderos. En la Figura 2 se ve un ejemplo de las imágenes a segmentar.

2. Metodología

2.1. Redes neuronales convolucionales (RNC)

Las *RNC* son algoritmos de aprendizaje profundo que procesan imágenes y entregan información espacial de las mismas [43]. Las redes neuronales simulan

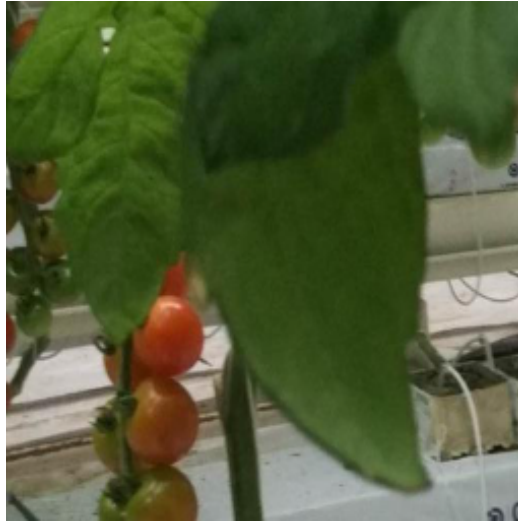


Fig. 2. Ejemplo de imagen a segmentar.

el sentido de la vista de los seres vivos, han sido desarrollados diferentes modelos con buenos resultados en las últimas dos décadas.

Los modelos desarrollados de *RNC* tienen en común que, en cada uno de ellos la cantidad de parámetros entrenables ha aumentado y la aparición de diferentes funciones de activación han posibilitado su uso en diferentes ámbitos y contextos, reportando buenos resultados en las tareas en que fueron probadas. En 1998 *Lenet* – 5 [19] es presentada con 60 mil parámetros, presenta por primera vez los kernel de convoluciones como funciones de activación y la agrupación de capas. En 2012 *AlexNet* [18] se presentó con 60 millones de parámetros e introdujo al mundo de las *RNC* las funciones de activación *ReLU*s y *Dropout*. VGG-16 [34] hace su aparición en el 2014 con 138 millones de parámetros, en el mismo año *Inception* – v1 [38] que fue la primera red que uso bloques de capas. Para el año 2015 el modelo *ResNet50* [14], con 44 millones de parámetros y popularizó la forma de realizar las conexiones actuales entre capas al disminuir el desvanecimiento del gradiente, logró 152 capas sin perder generalización en el descenso del gradiente y es de los modelos pioneros en usar "batch normalization".

El modelo de *RNC* *ResNet50* tiene gran relevancia en el mundo del aprendizaje profundo, debido a su capacidad de entrenar redes con gran profundidad en lo referente al número de capas interconectas [13], esto ha permitido ser aplicada en el de reconocimiento de imágenes [33], detección de objetos [24], reconocimiento de rostros [21,20,39] y clasificación de imágenes [21]. En el ámbito agrícola la *ResNet50* para la detección de plagas [9], detección de déficit nutrimentales en plantas [41].

El núcleo de este trabajo recae en una *RNC ResNet50* que es la encargada de realizar la tarea de segmentado semántico en imágenes de cultivos de jitomate. Las etapas para implementar el segmentador semántico mediante la *RNC ResNet50* para separar las hojas y frutos en imágenes con cultivos de jitomate, se muestran en la Figura 3. Cuenta con 6 etapas que son descritas a continuación:



Fig. 3. Metodología de implementación del segmentador.

1. Dataset.

Consta de 500 imágenes (ver ejemplo de las imágenes del dataset en la Figura 2), las que pertenecen a un dataset accesible para su descarga desde Internet [3]. Las imágenes del dataset tienen las siguientes características: formato de los archivos PNG, dimensiones de las imágenes ($N \times M$): 400×500 y 500×400 píxeles. Del conjunto de imágenes del dataset se crearon dos subconjuntos, de entrenamiento y prueba; para ello se realizó una asignación aleatoria de las 500 imágenes. El conjunto de entrenamiento quedó conformado por 350



Fig. 4. Etiquetado manual de la Figura 2.

imágenes, lo que corresponde al 70 % del dataset, mientras que el conjunto de prueba está integrado por las 150 imágenes restantes, lo que corresponde al 30 % de las 500 imágenes que conforman el dataset. Los porcentajes mencionados fueron determinados por la convención usada de 60 % para el conjunto de entrenamiento, el restante 40 % para prueba y validación. Al no tener un conjunto de validación se repartió equitativamente en los conjuntos de entrenamiento y prueba.

2. Etiquetado manual de las imágenes del dataset.

Para poder entrenar el segmentador semántico por medio de una *RNC ResNet50*, es necesario marcar o etiquetar los elementos que deseamos separar en las imágenes de los conjuntos de entrenamiento y de prueba, en nuestro caso: las hojas y frutos de cultivos de jitomate. El etiquetado de las imágenes se realizó con la herramienta "Computer Vision Annotation Tool" (CVAT) [1], la cual es una herramienta web gratuita. En CVAT se pueden realizar diferentes tipos de marcados o etiquetados para trabajar con diferentes algoritmos de aprendizaje profundo. En la Figura 4 se observa el etiquetado de los píxeles que corresponden a las hojas y frutos de la Figura 2, es importante mencionar que se realizó el etiquetado de las 500 imágenes del dataset, generando una máscara para cada imagen.

3. Redimensionar imágenes y máscaras.

Para poder realizar el entrenamiento del modelo de la *RNC ResNet50* es necesario transformar las imágenes y sus respectivas máscaras al tamaño de las entradas de nuestro modelo propuesto a utilizar.

4. Preparación del modelo.

Se establecen las dimensiones de las entradas de nuestro modelo y la cantidad de clases a segmentar, para el modelo propuesto serán de 256×256 las imágenes de entrada del conjunto de entrenamiento y clasificará 3 clases: hojas, frutos y fondo.

5. Entrenamiento del modelo de *RNC ResNet50*.

```

dim=(256,256)
img = cv2.imread(ruta_img)
imgr = cv2.resize(img, dim)
cv2.imwrite(archivo_destino,imgr)

```

Fig. 5. Ajuste de tamaño de imagen.

Se procede a realizar el entrenamiento del modelo propuesto con las 350 imágenes del conjunto de entrenamiento tomadas de aleatoriamente de las 500 imágenes del dataset..

6. Evaluación en la etapa de reconocimiento.

Con las imágenes del conjunto de prueba se evalúa la segmentación realizada por el modelo *RNC ResNet50* para medir su eficiencia en la tarea de separar las hojas y frutos de cultivos de jitomate.

3. Implementación

La implementación del modelo de *RNC ResNet50* se realizó con el lenguaje de programación de propósito general "Python", se usaron las librerías de keras [17] (<https://keras.io/>) para implementar el modelo de aprendizaje profundo y OpenCV [27] (<https://opencv.org/>) para la manipulación de las imágenes de los conjuntos de entrenamiento y prueba. Las líneas de código principales implementadas para la realización del segmentador semántico se describen a continuación:

1. Redimensionar imágenes del dataset y sus máscaras.

En la Figura 5 se muestran las líneas de código necesarias para cambiar las dimensiones de las imágenes de los conjuntos de entrenamiento y prueba a un tamaño específico, en este caso es de 256×256 píxeles. La tercera línea de código es la instrucción "cv2.resize" de "OpenCV", la cual recibe dos parámetros, el primero es la imagen a cambiar de tamaño y mientras que el segundo son las nuevas dimensiones que tendrá la imagen. Es necesario cambiar el tamaño de las imágenes para que coincidan con las dimensiones de la *RNC ResNet50*.

2. Binarizado de máscaras de las imágenes del dataset.

El binarizado de las máscaras es necesario para realizar el proceso de entrenamiento del modelo *RNC ResNet50*. El proceso de binarizado para las máscaras del conjunto de entrenamiento similares a la Figura 4 se realiza por medio del código que se muestra en la Figura 6.

3. Preparación del modelo de aprendizaje profundo de la *ResNet50*.

Para implementar el modelo de *RNC ResNet50* es necesario establecer los parámetros de configuración básicos, como lo son: 'n_classes', en este parámetro almacena la cantidad de clases a segmentar, 3 en este caso. Los parámetros dos y tres corresponden a las dimensiones de las imágenes con

```
img = cv2.imread(ruta_img)
mascara = np.zeros((ancho,alto,1))
for y in range(img.shape[0]):
    for x in range (img.shape[1]):
        if (img[y][x][1]>200):
            mascara[y][x]=1
        else:
            if (img[y][x][2]>200):
                mascara[y][x]=2
cv2.imwrite(archivo_destino_mascara,mascara)
```

Fig. 6. Código de binarizado de máscaras.

las que se entrenará el modelo *RNC*, dichos parámetros son asignados a "input_height" e "input_width".

La generación del modelo de *RNC ResNet50* se observa en el código de la Figura 7.

```
from keras_segmentation.models.unet import resnet50_unet

model = resnet50_unet(
    n_classes=3 ,
    input_height=256 ,
    input_width=256 )
```

Fig. 7. Generación del modelo *ResNet50*.

La Figura 8 muestra la adaptación de la *RNC ResNet50* para realizar el objetivo de segmentar las hojas y frutos en imágenes de cultivos de jitomate (ver Figura 2 para un ejemplo de las imágenes). La Figura 8 es generada usando la librería "visualkeras", las líneas de código para generar la Figura 8 se muestra en la Figura 9.

4. Entrenamiento del modelo.

Para realizar el entrenamiento del modelo generado en la Figura 7, se requiere asignar los parámetros: 'train_images', se le asigna la ruta en donde se localizan las imágenes del conjunto de entrenamiento. El parámetro "train_annotations" le es asignada la ruta de las máscaras de las imágenes del conjunto de entrenamiento. El tercer parámetro "checkpoints_path" se le asigna la ruta donde se almacenan los pesos después de cada iteración o época. El cuarto parámetro "epochs" le es asignado el número de épocas que durará el entrenamiento.

5. Evaluación del conjunto de imágenes de prueba.

En la Figura 11 se muestra la línea de código para realizar el proceso de segmentado de la imagen que se quiera separar sus hojas y frutos de cultivos de jitomate. Para realizar el proceso de segmentado semántico utilizado la *RNC ResNet50*, es necesario especificar en el parámetro "inp" la ruta a la imagen a segmentar y en el parámetro "out_fname" la ruta



Fig. 8. Modelo generado para el segmentador semántico.

```
import visualkeras
visualkeras.layered_view(model, to_file='output.png',
                          legend=True,)
```

Fig. 9. Código para generar imagen del modelo *ResNet50* de la Figura 8.

donde se almacena la imagen con los resultados del proceso de segmentación semántica.

4. Resultados

Los bloques de código mostrados en la etapa de **Implementación** desde la Figura 5 hasta la Figura 11, generan imágenes en las cuales se aprecian la segmentación semántica que se realizó por medio del modelo de *RNC ResNet50* de las hojas y frutos de las plantas de jitomate.

El modelo tuvo en eficiencia de 98 % en las etapa de entrenamiento y un 87 % en fase de prueba, dichos porcentajes corresponden a la métrica "Accuracy", observe la la Figura 12.

Las Figuras 13, 14 y 15 muestran el resultado de la segmentación realizada por el modelo descrito anteriormente en la sección de **Implementación**. En cada Figura se pueden apreciar 4 imágenes, la primera es la toma original del cultivo de jitomate, la segunda es la máscara de dicha toma, la tercera y cuarta es la segmentación semántica de las hojas y frutos respectivamente.

5. Conclusiones y trabajos futuros

El presente artículo reporta la primera etapa de un proyecto de agricultura de precisión, que es la segmentación de hojas y frutos en imágenes de cultivos de jitomate, de un sistema generador de soluciones nutritivas para el cultivo de la fruta antes mencionada. El modelo de *RNC ResNet50* descrito en las cuartillas anteriores, al tener un 87 % de eficacia con las imágenes del conjunto de pruebas, se considera como exitoso al momento realizar la segmentación semántica de las hojas y frutos de cultivos de jitomate. Un aspecto importante a considerar al momento de implementar una *RNC* para llevar a cabo un proceso de segmentado semántico, es la cantidad de tiempo que se requiere invertir para realizar el etiquetado de las imágenes. Otro factor a tomar en cuenta, es la poder de computo necesario para realizar el entrenamiento del modelo, ya que se requiere de tarjetas de vídeo para poder reducir los tiempos de entrenamiento.

```
model.train(  
    train_images = "ruta_imagen_entrenamiento",  
    train_annotations = "ruta_etiquetado_entrenamiento",  
    checkpoints_path = "ruta_pesos_intermedios",  
    epochs=25  
)
```

Fig. 10. Código para realizar el entrenamiento del modelo.

```
out = model.predict_segmentation(  
    inp=origen,  
    out_fname=destino)
```

Fig. 11. Evaluación de imágenes.

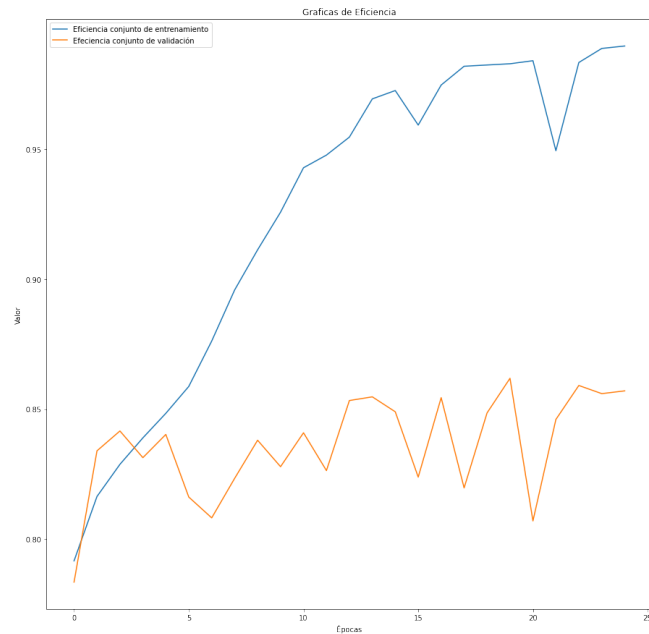


Fig. 12. Métricas de rendimiento en el entrenamiento "accuracy".

Las vertientes del trabajo futuro se dividen en: mejorar el rendimiento del segmentador semántico, llevar a cabo una comparativa de los resultados del segmentador semántico contra algún proceso de segmentación que implemente una técnica diferente y el desarrollo de un sistema use algún algoritmo heurístico para la creación de solución nutritiva que solvente la carencia detectada en las hojas o frutos segmentados.

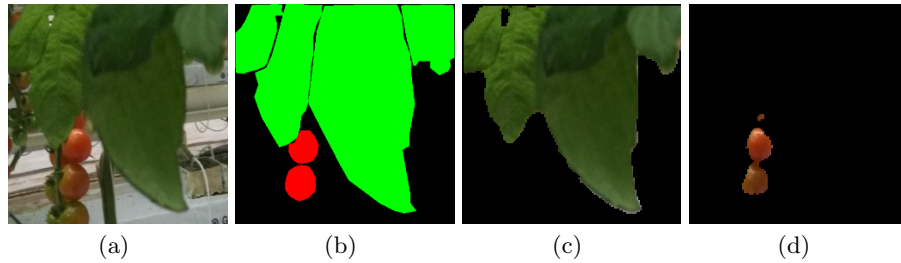


Fig. 13. Resultado del proceso de segmentación semántica ejemplo 1. a) Imagen Original b) máscara de la imagen c) Segmentación semántica de hojas d) Segmentación semántica de frutos

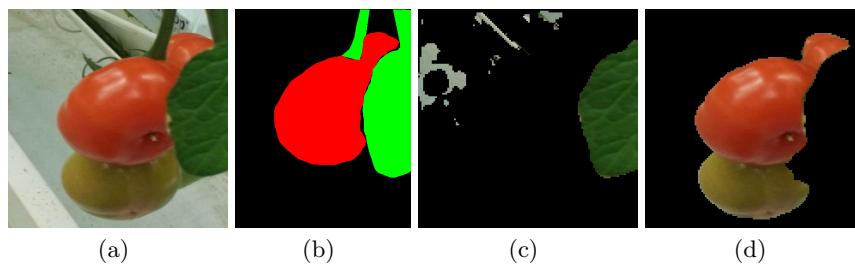


Fig. 14. Resultado del proceso de segmentación semántica ejemplo 2. a) Imagen Original. b) máscara de la imagen. c) Segmentación semántica de hojas. d) Segmentación semántica de frutos.



Fig. 15. Resultado del proceso de segmentación semántica ejemplo 3. a) Imagen Original. b) máscara de la imagen. c) Segmentación semántica de hojas. d) Segmentación semántica de frutos.

Para mejorar el desempeño del segmentador semántico mostrado en la Figura 12, se probará con las siguientes alternativas: aumentar la cantidad de épocas de entrenamiento, incrementar el número de imágenes en el conjunto de entrenamiento y mejorar el etiquetado de las imágenes del dataset.

En la segunda vertiente de trabajo se desarrollará un segmentador de hojas y frutos de cultivos de jitomates basado en umbrales con algún modelo de representación de color, como lo son el *RGB* y *HSV*, con los resultados de ambas segmentaciones, la semántica y por umbrales se realizará una comparativa de los resultados.

La tercera vertiente de trabajo, pretende con los resultados de la segmentación de las hojas y frutos de cultivos de jitomates se alimentará un detector de déficit de nutrientes, para posteriormente proponer una solución nutritiva que corrija el déficit detectado.

Agradecimientos. Es importante agradecer a las Instituciones y empresas que hicieron posible el desarrollo del presente trabajo al destinar recursos de diferentes índoles para la realización de de la misma:

- Consejo Nacional de Ciencia y Tecnología (CONACYT).
- Centro de Investigaciones en Óptica A.C.
- Instituto Tecnológico de Estudios Superiores de Zamora.
- Empresa Opus Farms.

Referencias

1. CVAT, <https://www.cvat.ai/>
2. File:ResNet50.png - Wikimedia Commons, <https://commons.wikimedia.org/wiki/File:ResNet50.png>
3. Tomato Detection — Kaggle, <https://www.kaggle.com/datasets/andrewmvd/tomato-detection>
4. Aruul Mozhi Varman, S., Baskaran, A. R., Aravindh, S., Prabhu, E.: Deep Learning and IoT for Smart Agriculture Using WSN. 2017 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2017, pp. 1–6 (2018) doi: 10.1109/ICCIC.2017.8524140
5. Badrinarayanan, V., Kendall, A., Cipolla, R., Badrinarayanan, V., Cipolla, R., Zhuang, J., Yang, J., Gu, L., Dvornek, N.: Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 12, pp. 2481–2495 (2017) doi: 10.5244/c.31.57
6. Bhargava, A., Bansal, A.: Fruits and vegetables quality evaluation using computer vision: A review. Journal of King Saud University - Computer and Information Sciences, vol. 33, no. 3, pp. 243–257 (2021) doi: 10.1016/j.jksuci.2018.06.002
7. Chen, L., Lin, L., Cai, G., Sun, Y., Huang, T., Wang, K., Deng, J.: Identification of nitrogen, phosphorus, and potassium deficiencies in rice based on static scanning technology and hierarchical identification method. PLoS ONE, vol. 9, no. 11, pp. 1–17 (2014) doi: 10.1371/journal.pone.0113200

8. Cilia, C., Panigada, C., Rossini, M., Meroni, M., Busetto, L., Amaducci, S., Boschetti, M., Picchi, V., Colombo, R.: Nitrogen status assessment for variable rate fertilization in maize through hyperspectral imagery. *Remote Sensing*, vol. 6, no. 7, pp. 6549–6565 (2014) doi: 10.3390/rs6076549
9. Fuentes, A., Yoon, S., Kim, S. C., Park, D. S.: A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors (Switzerland)*, vol. 17, no. 9 (2017) doi: 10.3390/s17092022
10. Garcia-Lamont, F., Cervantes, J., López, A., Rodriguez, L.: Segmentation of images by color features: A survey. *Neurocomputing*, vol. 292, pp. 1–27 (5 2018) doi: 10.1016/j.neucom.2018.01.091
11. Gebbers, R., Adamchuk, V. I.: Precision agriculture and food security. *Science*, vol. 327, no. 5967, pp. 828–831 (2010) doi: 10.1126/science.1183899
12. Hao, S., Zhou, Y., Guo, Y.: A Brief Survey on Semantic Segmentation with Deep Learning. *Neurocomputing*, vol. 406, pp. 302–321 (2020) doi: 10.1016/j.neucom.2019.11.118
13. He, K.: Deep Residual Learning for Image Recognition. *Indian Journal of Chemistry - Section B Organic and Medicinal Chemistry*, vol. 45, no. 8, pp. 1951–1954 (2006) doi: 10.1002/chin.200650130
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. vol. 2016-Decem, pp. 770–778. IEEE Computer Society (12 2016) doi: 10.1109/CVPR.2016.90
15. Jia, L., Chen, X., Zhang, F., Buerkert, A., Roemheld, V.: Optimum nitrogen fertilization of winter wheat based on color digital camera images. *Communications in Soil Science and Plant Analysis*, vol. 38, no. 11-12, pp. 1385–1394 (2007) doi: 10.1080/00103620701375991
16. Kamilaris, A., Prenafeta-Boldú, F. X.: Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, vol. 147, no. July 2017, pp. 70–90 (4 2018) doi: 10.1016/j.compag.2018.02.016
17. KERAS: Keras API reference / Optimizers / RMSprop (2021), <https://keras.io/api/>
18. Krizhevsky, A., Sutskever, I., Hinton, G. E.: ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, vol. 60, no. 6, pp. 84–90 (5 2017) doi: 10.1145/3065386
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323 (1998) doi: 10.1109/5.726791
20. Lu, Z., Jiang, X., Kot, A.: Deep Coupled ResNet for Low-Resolution Face Recognition. *IEEE Signal Processing Letters*, vol. 25, no. 4, pp. 526–530 (4 2018) doi: 10.1109/LSP.2018.2810121
21. Mandal, B., Okeukwu, A., Theis, Y.: Masked Face Recognition using ResNet-50. *CoRR*, vol. abs/2104.0 (4 2021)
22. Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D.: Image Segmentation Using Deep Learning: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542 (7 2022) doi: 10.1109/TPAMI.2021.3059968
23. Mohanty, S. P., Hughes, D., Salathe, M.: Using Deep Learning for Image-Based Plant Disease Detection. *Frontiers in Plant Science*, vol. 7, pp. 1–10 (4 2016) doi: 10.3389/fpls.2016.01419

24. Mukti, I. Z., Biswas, D.: Transfer Learning Based Plant Diseases Detection Using ResNet50. In: 2019 4th International Conference on Electrical Information and Communication Technology, EICT 2019. Institute of Electrical and Electronics Engineers Inc. (12 2019) doi: 10.1109/EICT48899.2019.9068805
25. Nanehkaran, Y. A., Zhang, D., Chen, J., Tian, Y., Al-Nabhan, N.: Recognition of plant leaf diseases based on computer vision. *Journal of Ambient Intelligence and Humanized Computing*, , no. 0123456789 (2020) doi: 10.1007/s12652-020-02505-x
26. Nyalala, I., Okinda, C., Nyalala, L., Makange, N., Chao, Q., Chao, L., Yousaf, K., Chen, K.: Tomato volume and mass estimation using computer vision and machine learning algorithms: Cherry tomato model. *Journal of Food Engineering*, vol. 263, no. July, pp. 288–298 (2019) doi: 10.1016/j.jfoodeng.2019.07.012
27. OpenCV: Home - OpenCV, <https://opencv.org/>
28. Patrício, D. I., Rieder, R.: Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and Electronics in Agriculture*, vol. 153, pp. 69–81 (10 2018) doi: 10.1016/j.compag.2018.08.001
29. Philipp, I., Rath, T.: Improving plant discrimination in image processing by use of different colour space transformations. *Computers and Electronics in Agriculture*, vol. 35, no. 1, pp. 1–15 (2002) doi: 10.1016/S0168-1699(02)00050-9
30. Pierce, F. J., Nowak, P.: Aspects of precision agriculture. *advances in Agronomy*, vol. 67, pp. 1–68 (1999)
31. Pokrajac, D., Obradovic, Z.: Neural network-based software for fertilizer optimization in precision farming. *Proceedings of the International Joint Conference on Neural Networks*, vol. 3, no. I, pp. 2110–2115 (2001) doi: 10.1109/ijcnn.2001.938492
32. Rorie, R. L., Purcell, L. C., Karcher, D. E., King, C. A.: The assessment of leaf nitrogen in corn from digital images. *Crop Science*, vol. 51, no. 5, pp. 2174–2180 (2011) doi: 10.2135/cropsci2010.12.0699
33. Sai Bharadwaj Reddy, D. Sujitha Juliet: Transfer Learning with ResNet-50 for Malaria Cell-Image Classification. *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*, pp. 945–949 (2019)
34. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings (9 2015), <http://arxiv.org/abs/1409.1556>
35. Singh, A. A. K. A., Ganapathysubramanian, B., Sarkar, S., Singh, A. A. K. A.: Deep Learning for Plant Stress Phenotyping: Trends and Future Perspectives. *Trends in Plant Science*, vol. 23, no. 10, pp. 883–898 (2018) doi: 10.1016/j.tplants.2018.07.004
36. Smith, R., Baillie, J., McCarthy, A., Raine, S. R., Baillie, C. P.: Review of precision irrigation technologies and their application. *National Centre for Engineering in Agriculture University of Southern Queensland Toowoomba*, , no. November (2010)
37. Story, D., Kacira, M., Kubota, C., Akoglu, A., An, L.: Lettuce calcium deficiency detection with machine vision computed plant features in controlled environments. *Computers and Electronics in Agriculture*, vol. 74, no. 2, pp. 238–243 (2010) doi: 10.1016/j.compag.2010.08.010
38. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception Architecture for Computer Vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2016-Decem, pp. 2818–2826 (12 2015) doi: 10.1109/CVPR.2016.308

39. Theckedath, D., Sedamkar, R. R.: Detecting Affect States Using VGG16, ResNet50 and SE-ResNet50 Networks. *SN Computer Science*, vol. 1, no. 2, pp. 1–7 (2020) doi: 10.1007/s42979-020-0114-9
40. Tian, H., Wang, T., Liu, Y., Qiao, X., Li, Y.: Computer vision technology in agricultural automation —A review. *Information Processing in Agriculture*, vol. 7, no. 1, pp. 1–19 (2020) doi: 10.1016/j.inpa.2019.09.006
41. Tran, T. T., Choi, J. W., Le, T. T. H., Kim, J. W.: A comparative study of deep CNN in forecasting and classifying the macronutrient deficiencies on development of tomato plant. *Applied Sciences (Switzerland)*, vol. 9, no. 8 (2019) doi: 10.3390/app9081601
42. Wang, Y., Wang, D., Zhang, G., Wang, J.: Estimating nitrogen status of rice using the image segmentation of G-R thresholding method. *Field Crops Research*, vol. 149, pp. 33–39 (2013) doi: 10.1016/j.fcr.2013.04.007
43. Wulandhari, L. A., Gunawan, A. A. S., Qurania, A., Harsani, P., Triastinurmiatiningsih, Tarawan, F., Hermawan, R. F.: Plant nutrient deficiency detection using deep convolutional neural network. *ICIC Express Letters*, vol. 13, no. 10, pp. 971–977 (2019) doi: 10.24507/icicel.13.10.971
44. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: BiSeNet: Bilateral segmentation network for real-time semantic segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11217 LNCS, pp. 334–349 (2018) doi: 10.1007/978-3-030-01261-8{-}20
45. Zhuang, J., Yang, J., Gu, L., Dvornek, N.: Fully Convolutional Networks for Semantic Segmentation. *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, pp. 847–856 (2019) doi: 10.1109/ICCVW.2019.00113