

SEGMENTACIÓN EN IMAGEN DE FRUTOS DE GRANADO USANDO DEEP LEARNING CON APLICACIÓN EN AGRICULTURA DE PRECISIÓN

Jaime Giménez-Gallego, Juan D. González-Teruel, Ana B. Toledo-Moreo, Manuel Jiménez-Buendía, Fulgencio Soto-Valles y Roque Torres-Sánchez

jaime.gimenez@upct.es, juando.gonzalez@upct.es, ana.toledo@upct.es, manuel.jimenez@upct.es,
pencho.soto@upct.es y roque.torres@upct.es

Grupo de investigación Sistemas de Precisión para la Sostenibilidad Agroalimentaria, Medioambiental y Social (PRISMA) Universidad Politécnica de Cartagena, Campus Muralla del Mar, s/n. 30202 Cartagena, España

Resumen

En agricultura de precisión, para monitorizar el estado del cultivo mediante imagen de forma automática, son necesarias herramientas de procesamiento para poder extraer la información de interés. En este estudio se desarrolla un modelo de Deep Learning para segmentación de imagen con el objetivo de discriminar los frutos del granado. Se alcanzan unos resultados de Intersection over Union (IoU)=0,71 y mean Average Precision (mAP)=0,82. Posteriormente, se expone un algoritmo que permite estimar el tamaño del fruto en píxeles, con un error relativo medio del 5,4%.

Palabras clave: Agricultura de Precisión; Visión por Computador; Deep Learning; Segmentación de Imagen

1 INTRODUCCIÓN

En agricultura de precisión resulta de interés monitorizar el estado del cultivo [1]. Así, se pueden tomar decisiones acerca de la gestión del riego o cosecha. Concretamente, si se trata de árboles frutales, hay que atender al estado del fruto. Existen trabajos relacionados con la temática que abordan el problema de la detección de fruto en entornos naturales para estimación de conteo para la cosecha o tamaño [2]–[17]. Esta metodología tiene su aplicación en seguimiento basado en vuelos de dron [2] o robots de recolección [5], [10]. Sin embargo, estas técnicas son de naturaleza puntual, con una resolución temporal reducida y de coste elevado.

Por esa razón, como alternativa, se postula la opción de una monitorización continua del fruto, basada en múltiples sensores de imagen repartidos por la plantación. Estos serían capaces de recabar información en los instantes deseados del día y ejecutar un seguimiento de la evolución temporal de los frutos presentes en la imagen. Para ello, es necesario contar con las herramientas que permitan

procesar la imagen original obtenida por el sensor de forma automática y disponer como resultado de parámetros de interés exclusivos del fruto.

El objetivo que se plantea en el estudio es ofrecer una solución a través de un modelo de inteligencia artificial, basado en Deep Learning (Aprendizaje Profundo), para la segmentación de imagen, concretamente segmentación de instancias, para discriminar los frutos del fondo y entre sí. Posteriormente, con las máscaras de las instancias obtenidas, relativas a los frutos, se pueden aplicar algoritmos para estimar el tamaño del fruto y seguir su evolución. En este estudio se trabaja específicamente con granados.

2 MATERIALES Y MÉTODOS

Para generar el modelo de visión artificial basado en Deep Learning, se recogió un conjunto de imágenes para el entrenamiento. La especie de árbol frutal objeto de la investigación fue el granado (*Punica granatum*). Las imágenes se obtuvieron por medio de dispositivos móviles en una finca de La Majada, Mazarrón, Murcia, España (37°37'40.3"N 1°24'03.8"W). Las imágenes presentaban una resolución original de 4624x3472 píxeles. El conjunto de datos se compuso por 213 imágenes en total. El procesamiento de datos y entrenamiento del modelo se realizaron mediante un ordenador con las siguientes características: CPU Intel® Core™ i7-10875H 8 núcleos @ 2.3GHz, RAM 32GB DDR4 @ 3200MHz, GPU GeForce® RTX 2070 8GB GDDR6. El lenguaje de programación empleado fue Python [18] y el entorno de programación Spyder [19]. Para la anotación manual de las imágenes de entrenamiento se hizo uso del programa SuperAnnotate [20].

En primer lugar, para poder entrenar el modelo de inteligencia artificial, es necesario obtener las salidas del conjunto de datos. En el caso de la segmentación de imagen, la entrada al modelo es la propia imagen y la salida es la máscara, que define la región de interés presente en la misma. Específicamente, en

segmentación de instancias, se distingue entre los diferentes elementos de la misma clase. Por tanto, la salida se conforma por una máscara para cada instancia, esto es, para cada fruto presente en la imagen. Así, de forma manual, para cada imagen del conjunto de datos, se marcó sobre la imagen original el área comprendida por cada fruto y se guardó la máscara (Figura 1).



Figura 1: Ejemplo de anotación de imagen para segmentación de instancias. La región de interés de cada fruto se marca en azul.

Con el fin de que el coste computacional del entrenamiento y posteriores predicciones fuera asumible, se redimensionaron las imágenes, así como las máscaras asociadas, a una resolución de 640x480 píxeles. El conjunto de datos se dividió en 3 grupos: conjunto de entrenamiento (183 imágenes), conjunto de validación (20 imágenes) y conjunto de prueba (10 imágenes). El conjunto de imágenes entrenamiento se destinó al aprendizaje del modelo para ajustarse al problema específico de discriminar los frutos del fondo y entre sí. El conjunto de validación se usó para evaluar el entrenamiento del modelo en curso y definir el criterio de parada dinámico. Este se basa en detener el entrenamiento cuando los resultados del conjunto de validación no mejoren durante varias iteraciones. La métrica empleada como indicador del comportamiento del modelo es el mAP (mean Average Precision). El conjunto de prueba se empleó para evaluar el modelo entrenado final y comprobar la eficacia. Las imágenes de este último conjunto se seleccionaron manualmente de forma que fueran altamente representativas, cubriendo las diferentes casuísticas del problema. Para el proceso de entrenamiento, se empleó la técnica de aumentado de datos. Esta consiste en aplicar pequeñas transformaciones geométricas y colorimétricas aleatorias (traslaciones, rotaciones, cambios de brillo y color) para generar imágenes artificiales. Así, en cada iteración de entrenamiento, la imagen que procesa el modelo no es idéntica, permitiendo un mayor recorrido sin caer en el sobreajuste (memorización de las imágenes). El

modelo inicial empleado es una implementación de Mask R-CNN [21], que cuenta con una arquitectura de red de tipo Feature Pyramid Network (FPN) y como backbone ResNet101.

Una vez entrenado el modelo segmentación de instancias, se diseñó un algoritmo que permitiera obtener un parámetro representativo del tamaño del fruto a partir de la máscara. Sin embargo, usualmente, las máscaras generadas por el modelo tenían geometrías complejas, alejadas de la forma del fruto del granado. Esto se debe a que, en condiciones reales, los frutos presentes en la imagen pueden aparecer ocluidos parcialmente por hojas, ramas o entre sí. Además, las granadas tienen una forma que se podría aproximar como esférica, si no fuera por la corona. Teniendo esto en consideración, para el algoritmo se planteó utilizar el radio del círculo ajustado al cuerpo de la granada como medida del tamaño del fruto. Para ello, primero era necesario filtrar la corona de la granada y aproximar a un círculo la máscara independientemente de que fuera incompleta. El procedimiento resultante del algoritmo es el siguiente: (1) se realiza la predicción con el modelo entrenado sobre la imagen de los frutos, (2) sobre la máscara obtenida para cada fruto se obtiene el contorno exterior, (3) se ajusta el círculo óptimo al contorno, (4) se rellena la máscara con el círculo ajustado, (5) se obtiene el nuevo contorno exterior de la máscara rellena, (6) se calcula la circularidad con la fórmula de Polsby-Popper (Ecuación 1), (7) se aplica la operación morfológica de apertura sobre la máscara rellena (con un kernel de forma circular y tamaño proporcional a la relación entre el área de la máscara y el de la imagen) para filtrar regiones de píxeles que sobresalen del cuerpo de la máscara que potencialmente se pueden corresponder a la corona, (8) se obtiene el contorno de la máscara tras aplicar la apertura, (9) se calcula la nueva circularidad, (10) se compara la circularidad de los contornos antes y después de la operación de apertura, (11) se clasifica la máscara en “Corona”/”No Corona” en función de la relación entre los valores de circularidad. (11.a) Si la circularidad ha mejorado sustancialmente (por encima de un umbral obtenido de forma experimental) se considera que había corona en la máscara, se ajusta el círculo al contorno tras apertura y el radio resultante en píxeles se asigna como radio del fruto. En cambio, (11.b) si no hay mejora suficiente en la circularidad, se considera que no había corona en la máscara, se obtiene el círculo envolvente al contorno previo a la apertura y el radio resultante en píxeles se asigna como radio del fruto.

$$Circularidad\ Polsby-Popper = \frac{4\pi A}{P^2} \quad (1)$$

siendo A = área y P = perímetro

3 RESULTADOS Y DISCUSIÓN

En cuanto al modelo de segmentación de imagen, se evaluó para obtener los resultados en el conjunto de prueba, compuesto por 10 imágenes. Para cada imagen, la predicción del modelo se representó gráficamente dejando en color la región clasificada como fruto y en escala de grises el fondo, así como mostrando los contornos de las bounding box, rectángulos que enmarcan cada una de las máscaras de las instancias, en verde (Figura 2).



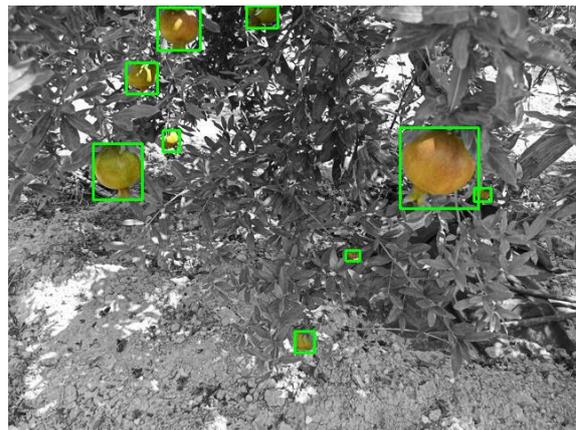
a)



b)



c)



d)

Figura 2: Ejemplos de predicción con el modelo de segmentación de instancias: a) imagen de prueba 1; b) predicción sobre la imagen de prueba 1; c) imagen de prueba 2; d) predicción sobre la imagen de prueba 2.

Así, tras realizar la predicción sobre las mismas, las máscaras de las instancias generadas automáticamente por el modelo entrenado se compararon con las anotaciones manuales correspondientes. Las métricas empleadas son: el IoU (Intersection over Union) y el mAP (mean Average Precision). Los resultados promedios calculados para todas las instancias de todas las imágenes del conjunto de prueba son: $\text{IoU} = 0,71$ y $\text{mAP} = 0,82$. Las desviaciones estándar asociadas a estos resultados son: $\text{IoU std} = 0,23$ y $\text{mAP std} = 0,15$.

Estos resultados se pueden considerar satisfactorios para la aplicación deseada. Sin embargo, en el contexto del estado del arte, un IoU de 0,71 es relativamente bajo al compararlo con los resultados de otros trabajos en los que se emplea la segmentación de instancias: $\text{IoU}=0,878$ con Mask-RCNN en manzana [5], $\text{IoU}=0,873$ con DaSNet-v2 en manzana [5] y $\text{IoU}=0,8985$ con Mask-RCNN en fresa [7]. Lo mismo ocurre con el mAP de 0,82: $\text{mAP}=0,9578$ con Mask-RCNN en fresa [7].

Se han observado casos en los que hay confusión en la predicción del modelo si dos frutos se solapan entre sí. Además, cuando un fruto se presenta ocluido parcialmente y una región del mismo queda al otro lado del elemento que lo divide, no se clasifica correctamente.

En lo relativo al algoritmo que permite obtener una medida del tamaño del fruto a partir de la máscara, se ha aplicado sobre 16 casos de prueba seleccionados manualmente que cubren el rango completo de las casuísticas observadas en las máscaras predichas por el modelo. Para cada máscara, se representa en verde el contorno exterior, en rojo el contorno de la circunferencia ajustada al mismo y su centro, y en

azul el centroide de la máscara. También se muestra, la máscara rellena tras la operación de apertura (en caso de que se clasifique como “Corona”) o la máscara rellena (en caso de que se clasifique como “No Corona”), representando en verde el contorno exterior de la máscara, en rojo el contorno de la circunferencia ajustada (“Corona”) o circunferencia envolvente (“No Corona”) y su centro, y en azul el centroide de la máscara (Figura 3).

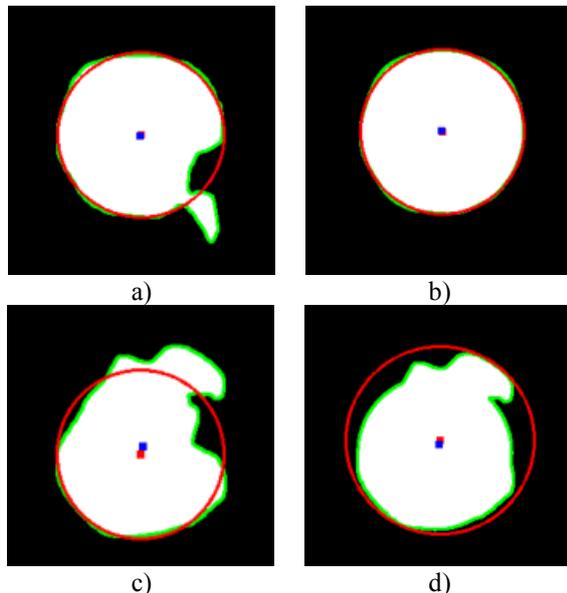


Figura 3: Ejemplos del algoritmo para obtener una estimación del radio del fruto a partir de la máscara: a) máscara 1 clasificada como “Corona”; b) máscara 1 clasificada como “Corona” rellena tras la operación de apertura; c) máscara 2 clasificada como “No Corona”; d) máscara 2 clasificada como “No Corona” rellena.

Sobre estos 16 casos de prueba, se ha obtenido un porcentaje de acierto en la clasificación “Corona”/“No Corona” de un 81%. El error relativo medio entre el radio de fruto predicho y el radio definido manualmente es del 5,4%. Si consideramos sólo aquellos casos en los que la clasificación del algoritmo ha resultado correcta, el error se reduce al 3,75%.

Por tanto, se considera necesario actuar en la mejora del algoritmo de clasificación, así como en el ajuste del círculo a la máscara original. Sin embargo, las imágenes presentan una complejidad elevada como para realizar una estimación altamente precisa del parámetro equivalente al tamaño únicamente en base a las máscaras segmentadas por el modelo de inteligencia artificial.

4 CONCLUSIONES

En el estudio presentado, se ha ofrecido una herramienta para segmentar en una imagen con

fondos naturales los frutos de granado y, posteriormente, poder estimar una medida de tamaño en píxeles. Sin embargo, al no contar con una referencia de dimensiones físicas conocidas en la imagen, resulta imposible obtener resultados de tamaño de fruto reales. Por esto, el método se limita a obtener medidas en píxeles, que sólo tienen sentido con carácter relativo, comparando el mismo fruto en diferentes momentos del proceso de maduración.

Como trabajos futuros, se plantea implementar esta metodología en una plantación sensorizada para monitorizar la evolución temporal del tamaño de los frutos de forma individualizada, reuniendo así información agronómica de interés. Además, resulta aplicable a otras especies de frutales.

Agradecimientos

Este trabajo ha sido financiado por la Agencia Estatal de Investigación (AEI) del Ministerio de Ciencia e Innovación (Convocatoria Retos investigación: Proyectos I+D+i 2017-2020) en el proyecto PID2019-106226RB-C2/AEI/10.13039/501100011033. El proyecto aborda la línea prioritaria relacionada con la conservación y gestión eficiente y sostenible de los recursos hídricos.

Además, se ha contado con la financiación del Ministerio de Ciencia, Innovación y Universidades: FPU17/05155, FPU19/00020 y EST21/00479.

English summary

IMAGE SEGMENTATION OF POMEGRANATE FRUITS USING DEEP LEARNING WITH APPLICATION IN PRECISION AGRICULTURE

Abstract

In precision agriculture, to automatically monitor the state of the crop using images, processing tools are needed to extract the information of interest. In this study, a Deep Learning model is developed for image segmentation to discriminate pomegranate fruits. Results of Intersection over Union (IoU)=0.71 and mean Average Precision (mAP)=0.82 are achieved. Subsequently, an algorithm for estimating the size of the fruit in pixels is presented, with an average relative error of 5.4%.

Keywords: Precision Agriculture; Computer Vision; Deep Learning; Image Segmentation

Referencias

- [1] L. P. Osco *et al.*, “Semantic segmentation of citrus-orchard using deep neural networks and multispectral UAV-based imagery,” *Precis. Agric.*, vol. 22, no. 4, pp. 1171–1188, Jan. 2021, doi: 10.1007/s11119-020-09777-5.
- [2] O. E. Apolo-Apolo, J. Martínez-Guanter, G. Egea, P. Raja, and M. Pérez-Ruiz, “Deep learning techniques for estimation of the yield and size of citrus fruits using a UAV,” *Eur. J. Agron.*, vol. 115, no. February, p. 126030, 2020, doi: 10.1016/j.eja.2020.126030.
- [3] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, “Apple detection during different growth stages in orchards using the improved YOLO-V3 model,” *Comput. Electron. Agric.*, vol. 157, pp. 417–426, Feb. 2019, doi: 10.1016/J.COMPAG.2019.01.012.
- [4] J. P. Vasconez, J. Delpiano, S. Vougioukas, and F. Auat Cheein, “Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation,” *Comput. Electron. Agric.*, vol. 173, p. 105348, Jun. 2020, doi: 10.1016/J.COMPAG.2020.105348.
- [5] H. Kang and C. Chen, “Fruit detection, segmentation and 3D visualisation of environments in apple orchards,” *Comput. Electron. Agric.*, vol. 171, p. 105302, Apr. 2020, doi: 10.1016/J.COMPAG.2020.105302.
- [6] H. Kang and C. Chen, “Fast implementation of real-time fruit detection in apple orchards using deep learning,” *Comput. Electron. Agric.*, vol. 168, p. 105108, Jan. 2020, doi: 10.1016/J.COMPAG.2019.105108.
- [7] Y. Yu, K. Zhang, L. Yang, and D. Zhang, “Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN,” *Comput. Electron. Agric.*, vol. 163, p. 104846, Aug. 2019, doi: 10.1016/J.COMPAG.2019.06.001.
- [8] K. Bresilla, G. D. Perulli, A. Boini, B. Morandi, L. Corelli Grappadelli, and L. Manfrini, “Single-shot convolution neural networks for real-time fruit detection within the tree,” *Front. Plant Sci.*, vol. 10, p. 611, Apr. 2019, doi: 10.3389/FPLS.2019.00611/BIBTEX.
- [9] C. Akin, M. Kirci, E. O. Gunes, and Y. Cakir, “Detection of the pomegranate fruits on tree using image processing,” *2012 1st Int. Conf. Agro-Geoinformatics, Agro-Geoinformatics 2012*, pp. 632–636, 2012, doi: 10.1109/AGRO-GEINFORMATICS.2012.6311724.
- [10] W. Jia, Y. Tian, R. Luo, Z. Zhang, J. Lian, and Y. Zheng, “Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot,” *Comput. Electron. Agric.*, vol. 172, p. 105380, May 2020, doi: 10.1016/j.compag.2020.105380.
- [11] G. Lin, Y. Tang, X. Zou, J. Cheng, and J. Xiong, “Fruit detection in natural environment using partial shape matching and probabilistic Hough transform,” *Precis. Agric.*, vol. 21, no. 1, pp. 160–177, 2019, doi: 10.1007/s11119-019-09662-w.
- [12] A. Koirala, K. B. Walsh, Z. Wang, and C. McCarthy, “Deep learning – Method overview and review of use for fruit detection and yield estimation,” *Comput. Electron. Agric.*, vol. 162, pp. 219–234, Jul. 2019, doi: 10.1016/J.COMPAG.2019.04.017.
- [13] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, “DeepFruits: A Fruit Detection System Using Deep Neural Networks,” *Sensors*, vol. 16, no. 8, p. 1222, Aug. 2016, doi: 10.3390/S16081222.
- [14] S. Bargoti and J. Underwood, “Deep fruit detection in orchards,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 3626–3633, Jul. 2017, doi: 10.1109/ICRA.2017.7989417.
- [15] A. Koirala, K. B. Walsh, Z. Wang, and C. McCarthy, “Deep learning for real-time fruit detection and orchard fruit load estimation: benchmarking of ‘MangoYOLO,’” *Precis. Agric.*, vol. 20, no. 6, pp. 1107–1135, Dec. 2019, doi: 10.1007/S11119-019-09642-0/TABLES/10.
- [16] S. Bargoti and J. P. Underwood, “Image Segmentation for Fruit Detection and Yield Estimation in Apple Orchards,” *J. F. Robot.*, vol. 34(6), pp. 1039–1060, 2017, doi: 10.1002/rob.21699.
- [17] S. Wan and S. Goudos, “Faster R-CNN for multi-class fruit detection using a robotic vision system,” *Comput. Networks*, vol. 168, p. 107036, Feb. 2020, doi: 10.1016/J.COMNET.2019.107036.
- [18] “Python official website.” <https://www.python.org/> (accessed Dec. 11, 2019).
- [19] “Home — Spyder IDE.” <https://www.spyder-ide.org/> (accessed Jun. 30, 2022).
- [20] “The ultimate training data platform for AI | SuperAnnotate.” <https://www.superannotate.com/> (accessed Jun. 30, 2022).
- [21] “GitHub - matterport/Mask_RCNN: Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow.” https://github.com/matterport/Mask_RCNN

(accessed Jun. 29, 2022).



© 2022 by the authors.
Submitted for possible open access publication
under the terms and conditions of the Creative
Commons Attribution CC-BY-NC-SA 4.0 license
(<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).