

<https://doi.org/10.23913/ride.v14i27.1770>

Artículos científicos

Aprendizaje para obtener la medición del costo computacional de los algoritmos de reconocimiento de imágenes en hojas del cultivo de soya

Learning to obtain the measure of computational cost of the image recognition algorithms in leaves of the soybean crop

Aprendendo a medir o custo computacional de algoritmos de reconhecimento de imagem em folhas de soja

Felipe Anastacio González González

Universidad Autónoma de Tamaulipas, México.

felgonzale@docentes.uat.edu.mx

<https://orcid.org/0000-0002-1410-8616>

Joaquín Torres Mata

Universidad Autónoma de Tamaulipas, México

jtorresma@docentes.uat.edu.mx

<https://orcid.org/0000-0002-9298-8831>

Angel Mario Lerma Sánchez

Universidad Autónoma de Tamaulipas, México.

amlerma@docentes.uat.edu.mx

<https://orcid.org/0000-0002-5759-1302>

Irma Carolina González Sánchez

Universidad Autónoma de Tamaulipas, México.

carolinagonzalez327@gmail.com

<https://orcid.org/0000-0003-2745-0178>

Resumen

El objetivo de este trabajo fue presentar la medición del costo computacional de varios algoritmos de reconocimiento de imágenes en hojas de soya, a través de un enfoque cuantitativo y experimental. Después de revisar la base teórica, se procedió a la implementación y experimentación de estos algoritmos utilizando imágenes de hojas de soya. Los resultados demuestran que no se puede establecer claramente cuál es la mejor elección entre los algoritmos detectores de descriptores como SIFT y SURF, pues mientras uno de ellos tiene un mayor tiempo de procesamiento en milisegundos, su consumo de memoria es menor, y viceversa. Por otro lado, en cuanto a los algoritmos de búsqueda de esquinas, como Harris y Shi-Tomasi, este último demostró ser superior tanto en términos del número de córneres detectados como en el tiempo de procesamiento en milisegundos y el consumo de memoria. Finalmente, en el grupo de algoritmos de contornos activos, se observó que el algoritmo Snake supera al Chan-Vese con un menor tiempo de procesamiento en milisegundos y un menor consumo de memoria. En síntesis, se puede sugerir que, para el reconocimiento de hojas en plantas de soya, el algoritmo Shi-Tomasi podría ser una opción adecuada debido a su desempeño óptimo en cuanto al tiempo de procesamiento y al consumo de memoria en comparación con los otros algoritmos analizados.

Palabras clave: algoritmos, reconocimiento de imágenes, costo computacional.

Abstract

The objective of this paper is to present the measurement of the computational cost of some image recognition algorithms in soybean leaves based on the experimentation of these algorithms. It is a quantitative and experimental study. Starting from a search of the theoretical foundation of the algorithms, the implementation and experimentation of these algorithms tested with images of soybean crop leaves was carried out.

As for the results obtained, it is found that between the descriptor detection algorithms SIFT and SURF in obtaining descriptors for each image tested there is no clear choice because while one has more processing time in milliseconds, its memory consumption is lower and vice versa. In relation to the Harris and Shi-Tomasi corner search algorithms, there is one that clearly shows that it is better both in the number of corners detected, as well as the time in milliseconds is less and the memory consumption is also lower, in this case it is the Shi-Tomasi. And among the active contour algorithms we have that both the Snake and Chan-

Vese algorithms, among them the one with the best response time is the Snake with less time in milliseconds and less memory consumption.

Summarizing the results, it can be suggested that the Shi-Tomasi algorithm for the recognition of leaves in soybean plants would be adequate since it obtains optimal results in terms of time and memory consumption of the computer equipment in comparison with the other mentioned algorithms.

Keywords: Algorithms, image recognition, computational cost.

Resumo

O objetivo deste trabalho foi apresentar a mensuração do custo computacional de diversos algoritmos de reconhecimento de imagens em folhas de soja, através de uma abordagem quantitativa e experimental. Após revisão da base teórica, procedemos à implementação e experimentação desses algoritmos utilizando imagens de folhas de soja. Os resultados mostram que não se pode estabelecer claramente qual a melhor escolha entre algoritmos detectores de descritores como SIFT e SURF, pois embora um deles tenha maior tempo de processamento em milissegundos, seu consumo de memória é menor, e vice-versa. Por outro lado, no que diz respeito aos algoritmos de busca de cantos, como Harris e Shi-Tomasi, este último mostrou-se superior tanto no número de cantos detectados como no tempo de processamento em milissegundos e no consumo de memória. Por fim, no grupo de algoritmos de contorno ativo, observou-se que o algoritmo Snake supera o Chan-Vese com menor tempo de processamento em milissegundos e menor consumo de memória. Em resumo, pode-se sugerir que, para reconhecimento de folhas em plantas de soja, o algoritmo Shi-Tomasi poderia ser uma opção adequada devido ao seu ótimo desempenho em termos de tempo de processamento e consumo de memória em comparação aos demais algoritmos analisados.

Palavras-chave: algoritmos, reconhecimento de imagens, custo computacional.

Fecha Recepción: Febrero 2023

Fecha Aceptación: Septiembre 2023

Introducción

La soya es un arbusto tipo hierba, vertical, anual y con ramas, cuya altura varía entre los 30 centímetros y los 2 metros y su ciclo de vida puede oscilar entre los 80 y los 200 días, dependiendo de la variedad y las condiciones ambientales. Tanto las vainas, los tallos como las hojas de esta planta son pubescentes, es decir, están cubiertos de vellosidades. Además, sus semillas son de forma esférica, de tamaño medio y contienen niveles elevados de aceites y proteínas. En variedades mejoradas de soya, estas cifras pueden alcanzar hasta un 40 % de proteínas y un 22 % de aceite en relación con su peso en seco (Rosas y Young, 1991). La soya, de hecho, es el cultivo oleaginoso más extenso del mundo, pues existen alrededor de 130 millones de hectáreas de tierra que producen un total de 360 millones de toneladas (Roján-Herrera *et al.*, 2022).

Sin embargo, como cualquier otro cultivo, la soya es vulnerable a las plagas —como los deshojadores, hijuelos y raspadores (Magallanes-Estala *et al.*, 2014)—, y el daño causado por los insectos puede tener un impacto significativo en el rendimiento y la calidad de la cosecha. La magnitud de esta amenaza varía de un periodo de cultivo a otro, dependiendo de factores como el entorno, el tipo de insectos presentes, la ubicación geográfica y las condiciones climáticas. Por lo tanto, se recomienda llevar a cabo inspecciones visuales periódicas para detectar posibles infestaciones (Magallanes-Estala *et al.*, 2014).

Ahora bien, en lo que respecta al reconocimiento de imágenes, en el contexto de la visión artificial, se puede definir como “la capacidad del *software* para identificar objetos, lugares, personas, escritura y acciones en imágenes. Las computadoras pueden usar tecnologías de visión artificial en combinación con una cámara y un *software* de inteligencia artificial para lograr el reconocimiento de imágenes” (Ramírez-González *et al.*, 2009).

En el caso de las hojas de soya, conocer cuál es el mejor algoritmo para su reconocimiento no solo impacta directamente en la productividad agrícola, sino que también puede impulsar el progreso tecnológico, la sostenibilidad y la eficiencia a través de soluciones informáticas que beneficien a este sector. Por eso, el propósito del presente trabajo es evaluar el costo computacional de los algoritmos SIFT, SURF, Harris, Shi-Tomasi, Snake y Chan-Vese mediante su comparación en el proceso de reconocimiento de imágenes de plantas de soya. Para eso, se plantea la siguiente hipótesis:

- El costo computacional varía al utilizar diferentes algoritmos de reconocimiento de imágenes para la detección de hojas de plantas de soya.

En tal sentido, la pregunta de investigación formulada fue la siguiente:

- ¿Cuál es el costo computacional de los algoritmos SIFT, SURF, Harris, Shi-Tomasi, Snake y Chan-Vese en el reconocimiento de imágenes de plantas de soya?

Metodología

En la primera sección del apartado de metodología se describen los trabajos previos relacionados con el reconocimiento de hojas de soya. En la segunda sección se establecen las definiciones clave necesarias para una comprensión completa del artículo. A continuación, en la tercera sección, se presenta algunos algoritmos de reconocimiento de imágenes que pueden ser aplicados en el cultivo de la soya. Por último, en la cuarta sección se describen los experimentos realizados.

Trabajos relacionados

En cuanto a los antecedentes, Larese *et al.* (2011) desarrollaron una metodología que utiliza el análisis de imágenes de soya para detectar la cantidad de clorofila en las plantas, lo que sirve como indicador de la calidad fisiológica de las semillas. Este enfoque, además, permite la clasificación e identificación de diferentes variedades de cultivos de soya mediante un sistema automatizado de reconocimiento de imágenes foliares.

En otro estudio —realizado por Cáceres-Flores *et al.* (2015)— se menciona que existen trabajos especializados en el procesamiento de imágenes para la detección de plagas y el análisis morfológico de las plantas en cultivos de invernadero. Según los autores, se pueden detectar diferencias de color entre las hojas de plantas sanas e infectadas para identificar potencialmente la presencia de plagas a través de la segmentación de imágenes.

Además, se ha indagado en la detección temprana de plagas en plantas de pimiento utilizando el modelo de color RGB para procesar imágenes y examinar las características de las hojas y tallos. De esta manera se pueden identificar plagas agrupando las imágenes por color en función de los síntomas de enfermedad o infestación (Canty, 2019).

Asimismo, se ha implementado un sistema de detección de plagas mediante el procesamiento de imágenes en el cultivo de caña de azúcar (*Begonia semperflorens*). Específicamente, Lin *et al.* (2017) han investigado sobre el uso de drones para la inspección fotográfica de la cosecha de caña de azúcar, lo que representa una valiosa contribución a este campo.

Definiciones

Según la Real Academia de Ingeniería (s. f.), el costo computacional se refiere al valor en memoria y tiempo de ejecución de un sistema de información, aplicación, etc., a nivel de *hardware*, *software* y mantenimiento. Por otro lado, para la Real Academia Española (s. f.) una imagen (del latín *imago*) es una representación visual que muestra el aspecto de un objeto real o imaginario. Existen dos categorías principales de imágenes: las “mentales”, que las personas no perciben, pero existen en su interior, y las “creadas” o “repetidas”, que representan visualmente un objeto a través de diversos métodos, como dibujos, pinturas, fotografías o videos.

En un contexto más especializado, una imagen se define generalmente como una función $f(x, y)$ bidimensional que representa la intensidad de la luz, donde “x” e “y” representan las coordenadas espaciales de un punto, mientras que $f(x, y)$ es la intensidad o el nivel de gris de la imagen en ese punto. Cuando las coordenadas espaciales son finitas y la amplitud cubre un rango finito de valores, se denomina “imagen digital” (Sánchez-Prieto, 2014), la cual es una representación bidimensional de una imagen que utiliza bits, es decir, las unidades más pequeñas de información constituidas por dígitos binarios (0 y 1). Las computadoras y otros dispositivos digitales emplean imágenes digitales, que se pueden definir matemáticamente como una función bidimensional $f(x, y)$, donde “f” y los valores de (x, y) son cantidades discretas finitas que representan un valor de píxel específico dentro de la imagen (Peña-Peñate *et al.*, 2016).

Una imagen digital se puede representar mediante una matriz $f(x, y)$ de tamaño $M \times N$ de la siguiente manera:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0, N-1) \\ f(1,0) & f(1,1) & f(1, N-1) \\ f(M-1,0) & f(M-1,1) & f(M-1, N-1) \end{bmatrix}$$

Los elementos de la matriz (píxeles), en una imagen monocromática típica (que opera en niveles de gris), tienen 2, 8 o 256 de intensidad, por lo que $f(x, y)$ puede variar de 0 a 255 (0 es negro y 255 significa blanco); esto es lo que se puede representar como un carácter en la mayoría de los lenguajes de programación.

Un algoritmo, en cambio, se define como un conjunto de pasos que, cuando se realizan correctamente, conducen a un resultado (Gómez-Fuentes *et al.*, 2014). Según el reconocimiento de imágenes, se refiere a la clasificación de diferentes objetos de imagen en

función de sus descriptores. Los objetos detectados con descriptores similares se agrupan en una clase (Fernández-García, 2012). El costo computacional de un algoritmo se refiere a la cantidad de tiempo requerido para completar alguna operación; de igual manera, también tiene que ver con la memoria RAM utilizada para realizar dicha tarea (Trefethen y Bau, 1997).

Explicado lo anterior, es importante ofrecer un listado y una comparativa de algoritmos para medir su costo computacional, lo cual permitiría elegir la mejor opción según el costo y la confiabilidad.

Algoritmos de reconocimiento de imágenes

En el campo de la visión artificial uno de los problemas clásicos es el del reconocimiento o clasificación de imágenes, que consiste básicamente en tomar una o un conjunto de imágenes como entradas para generar como salida una etiqueta apropiada para dichas imágenes. En el caso de las hojas de soya, el objetivo es predecir si se trata de una hoja de soya, si tiene plaga y qué tipo de plaga presenta.

Para llevar a cabo esa tarea, el primer paso consiste en investigar y probar diversos algoritmos que ayuden en su ejecución. En el campo de la visión artificial, existen numerosas técnicas que varían según los objetivos que se deseen alcanzar. Algunas de ellas son las siguientes: métodos basados en características que detectan esquinas y manchas, y vectores descriptivos que se extraen alrededor de la vecindad de esos puntos. Asimismo, existen enfoques de aprendizaje automático mediante los cuales se aprenden las características de los datos de entrenamiento, es decir, dependiendo del problema se elige la técnica (Le, 12 de abril de 2018; Mallick, 2016).

Listado de algoritmos de reconocimiento de patrones

Los algoritmos vinculados con la visión por computadora y el reconocimiento de patrones son muy diversos y engloban muchas técnicas y objetivos (Rivera-Díez, 2015). A continuación, se procederá a nombrar y explicar diferentes algoritmos de detección de características en una imagen.

SIFT (Scale Invariant Features Transform)

El SIFT (Scale Invariant Features Transform) es un algoritmo de visión por computadora —difundido originalmente en 1999 por David Lowe— que se encarga de seleccionar características sobresalientes en imágenes en escala de grises. A partir de estas, se puede identificar una imagen en la base de datos y otra más grande con un número diferente de elementos dispuestos de manera desordenada (Ñauñay-Ilbay y Tipantuña-Córdova, 2013). El algoritmo descrito por Lowe consta de cuatro etapas:

- Detección de límites en el espacio-escala: En esta etapa se buscan puntos de interés para toda la imagen y se consideran todas las escalas utilizando la diferencia gaussiana.
- Ubicación precisa del punto clave: Para cada punto de interés anterior el modelo se ajusta para definir su ubicación y escala. Asimismo, se eligen puntos característicos (puntos clave), excepto aquellos que están cerca de los bordes o tienen poco contraste.
- Asignación de dirección: A cada punto base se le asigna una o más direcciones en función de la dirección del gradiente local. Esta orientación, junto con la posición y la escala previamente calculada, permite que el descriptor sea invariante en estos tres casos.
- Descripción de los puntos clave: Los gradientes de imagen locales se miden alrededor de cada punto característico y su histograma se usa para obtener una imagen de esa región que es resistente a cambios de luz significativos y pequeñas distorsiones de forma (Alegre y Fernández-Robles, 2016).

SURF (Speed Up Robust Feature)

El algoritmo SURF (Speed Up Robust Feature) es otro detector de variables locales. Fue presentado por primera vez por Bay *et al.* (2006) y se basa en el descriptor SIFT, aunque presenta algunos progresos como los siguientes:

- Mayor velocidad de cálculo sin deterioro de rendimiento.
- Cuando existe una transformación de la imagen presenta una mayor firmeza.

Los avances en esta área son factibles porque permiten la disminución de la complejidad del cálculo y la dimensionalidad de los vectores de características de puntos de interés conseguidos, aunque sigue siendo distinto y repetitivo (Aracil-López, 2012). A continuación, se listan las etapas en las que se divide:

- Localización de puntos de interés o *keypoints*.
- Fijación de la orientación.

- Extracción de los descriptores.

Harris Corner Detector

El detector de Harris se funda en la búsqueda de esquinas. Estas características son muy insensibles a los cambios de rotación y escala. Las esquinas (o córneres) son regiones en una imagen con variaciones de intensidad en diferentes direcciones, lo que representa el fundamento para encontrar el punto de Harris. Filtrando la imagen con una ventana móvil en ocho direcciones, se obtienen tres tipos de región (Enebral-González, 2009). A continuación, se indican las etapas en las que se divide:

- Para cada píxel (x, y) calcular la matriz de autocorrelación.
- Construir el mapa de variaciones para cada píxel (x, y).
- Se aplica umbralización al mapa de intensidades.
- Realizar supresión de no-máximos para encontrar máximos locales.

Shi-Tomasi Cornes Detector

En 1994 J. Shi y C. Tomasi realizaron pequeñas modificaciones y consiguieron mejores resultados que con el Harris Corner Detector. Este algoritmo propone los siguientes cambios respecto al anterior (Rivera-Díez, 2015).

$$R = \min (\lambda_1 \lambda_2)$$

Si el valor obtenido es mayor al valor umbral, significa que se encuentran en una esquina. Si se traza de nuevo el dibujo esquemático, quiere decir que se halla con la siguiente figura, en la que se puede ver a simple vista que solamente se considera esquina cuando los valores λ_1 y λ_2 se encuentran ambos por encima de λ_{min} .

Listado de algoritmos de contornos activos

Los modelos de contorno activos con representación semiautomática de objetos en imágenes intentan modelar las funciones energéticas asociadas a estas fuerzas (externas e internas) para desarrollar curvas guiadas por fuerzas externas y por límites y fuerzas asociadas a la imagen. Por lo general, estos modelos se basan en la detección de bordes utilizando información recopilada de los gradientes de imagen (Jiménez-Carretero *et al.*, 2011).

Contornos activos (Snakes)

En 2000, Ntalianis *et al.* escribieron un algoritmo para conseguir automáticamente los contornos iniciales de las Snakes a partir de mapas de profundidad obtenidos utilizando un algoritmo recursivo de árbol de expansión más corto (RSST) de múltiples resoluciones (Rodríguez-Rieiro, 2011). El algoritmo, una vez que se conoce la posición z del objeto de interés, devuelve una imagen de tres colores:

- Blanco: Si el objetivo de análisis corresponde a una profundidad diferente a la profundidad del levantamiento.
- Gris: Cuando el objetivo de análisis está cerca de la profundidad de investigación.
- Negro: Cuando el contenido que se analiza corresponde a la profundidad de la investigación.

El contorno inicial está formado por los puntos que pertenecen al borde y están más cerca del límite entre las partes gris y blanca de la imagen anterior (Rodríguez-Rieiro, 2011).

Algoritmo Chan-Vese

El método de Chan-Vese ha sido de gran utilidad para lograr la segmentación de objetos o diferentes estructuras anatómicas en imágenes con una gran cantidad de ruido o donde los bordes de los objetos de interés no están definidos. Hay trabajos en la literatura que dividen en grupos las estructuras anatómicas (por ejemplo, el intestino delgado y los vasos sanguíneos), así como en áreas industriales para la producción de carburadores, llantas y otras partes de automóviles. En general, el algoritmo de Chan-Vese es un método de segmentación que tiene como objetivo minimizar la función de energía (Hernández-Juárez *et al.*, 2017).

Experimentación

Para medir la efectividad de los algoritmos seleccionados se realizaron una serie de pruebas de acuerdo con el tipo de algoritmo, las cuales se ejecutaron con el siguiente ambiente de desarrollo: un equipo de cómputo que contaba con un Intel Core i5 7200U a 2.5 Ghz de procesador y 8 GB de memoria RAM sobre un Windows 10 de 64 bits como sistema operativo; en cuanto a las imágenes, la imagen 1 (principal) fue de 5184 x 2912 pixeles y 2.1 MB (figura 1), y la imagen 2 fue de 336 x 408 pixeles de 25.1 kB (figura 2); lo anterior se procesó usando el lenguaje de programación Python (versión 3. 6. 6) y el

paquete OpenCV (versión 3. 4. 2).

Figura 1. Imagen 1 (principal) de 5184 x
2912 pixeles



Fuente: Elaboración propia

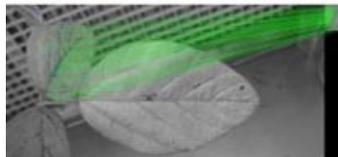
Figura 2. Imagen 2 de 336 x 408 pixeles



Fuente: Elaboración propia

Para la implementación del algoritmo SIFT los resultados se pueden observar en la figura 3, donde se obtuvo 1377 y 263 descriptores por cada imagen respectivamente (figuras 1 y 2). Estas se procesaron en un tiempo de 8782.877 milisegundos, con un consumo de memoria RAM de 46.16 MB (figura 3).

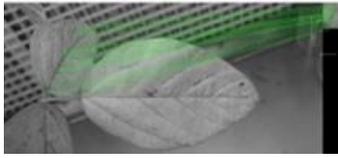
Figura 3. Resultado algoritmo SIFT



Fuente: Elaboración propia

En cuanto al algoritmo SURF, se consiguieron 2553 y 153 descriptores por cada imagen respectivamente (figura 1 y 2), con un tiempo de 7004.71 milisegundos y un consumo de memoria RAM de 65.75 MB (figura 4).

Figura 4. Resultado algoritmo SURF



Fuente: Elaboración propia

En cuanto al algoritmo de Harris, se detectaron 46 esquinas en un tiempo de 3837.095 milisegundos, con un consumo de memoria RAM de 276.894 MB.

Figura 5. Resultado algoritmo Harris



Fuente: Elaboración propia

Referente al algoritmo de Shi-Tomasi, se destacaron 1000 esquinas en un tiempo de 3688.562 milisegundos con un consumo de memoria de 104.417 MB.

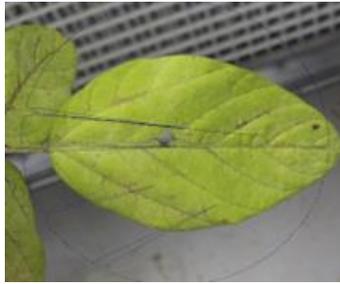
Figura 6. Resultado Algoritmo Shi-Tomasi



Fuente: Elaboración propia

En relación con los algoritmos de contornos activos, Snake arrojó un tiempo de ejecución de 40 segundos y un consumo de memoria de 394.199 MB (figura 7).

Figura 7. Resultado algoritmo Snake



Fuente: Elaboración propia

En cuanto al algoritmo de Chan-Vese, se tuvo un tiempo de ejecución de 7.7 minutos y un consumo de memoria de 1890.929 MB.

Figura 8. Resultado algoritmo Chan-Vese



Fuente: Elaboración propia

Resultados

La tabla 1 resume los resultados alcanzados con los algoritmos SIFT y SURF.

Tabla 1. Resultados de los algoritmos SIFT y SURF

| DESCRIPTORES | | | | |
|--------------|--------|--------|--------------|--------------------|
| Algoritmo | Fig. 1 | Fig. 2 | Tiempo en ms | Consumo de memoria |
| SIFT | 1377 | 263 | 8782.877 | 46.16 |
| SURF | 2553 | 153 | 7004.71 | 65.75 |

Fuente: Elaboración propia

La tabla 2 muestra los resultados obtenidos con los algoritmos Harris y Shi-Tomasi.

Tabla 2. Resultados de los algoritmos Harris y Shi-Tomasi

| PROCESAMIENTO | | | |
|---------------|-----------------|--------------|--------------------|
| Algoritmo | Num. De Corners | Tiempo en Ms | Consumo de memoria |
| Harris | 46 | 3837.095 | 276.894 |
| Shi-Tomasi | 1000 | 3688.562 | 104.417 |

Fuente: Elaboración propia

La tabla 3 refleja los resultados obtenidos en los algoritmos Snake y Chav-Vesse.

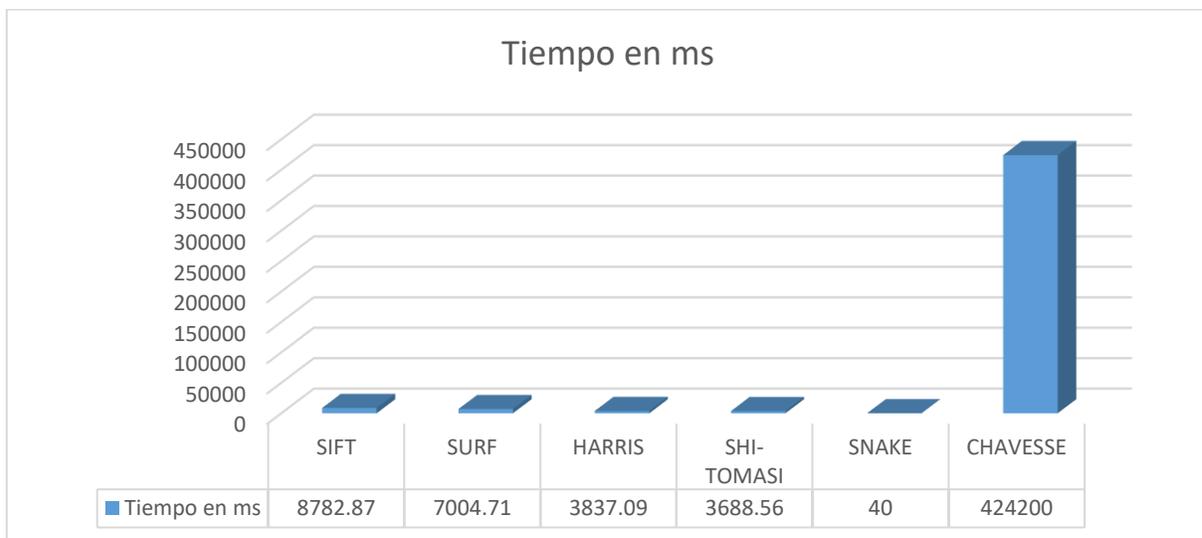
Tabla 3. Resultados de los algoritmos Snake y Chav-Vesse

| Algoritmo | Tiempo ms | Consumo de memoria |
|-----------|-----------|--------------------|
| Snake | 40 | 394.199 |
| Chan-Vese | 424200 | 1890.929 |

Fuente: Elaboración propia

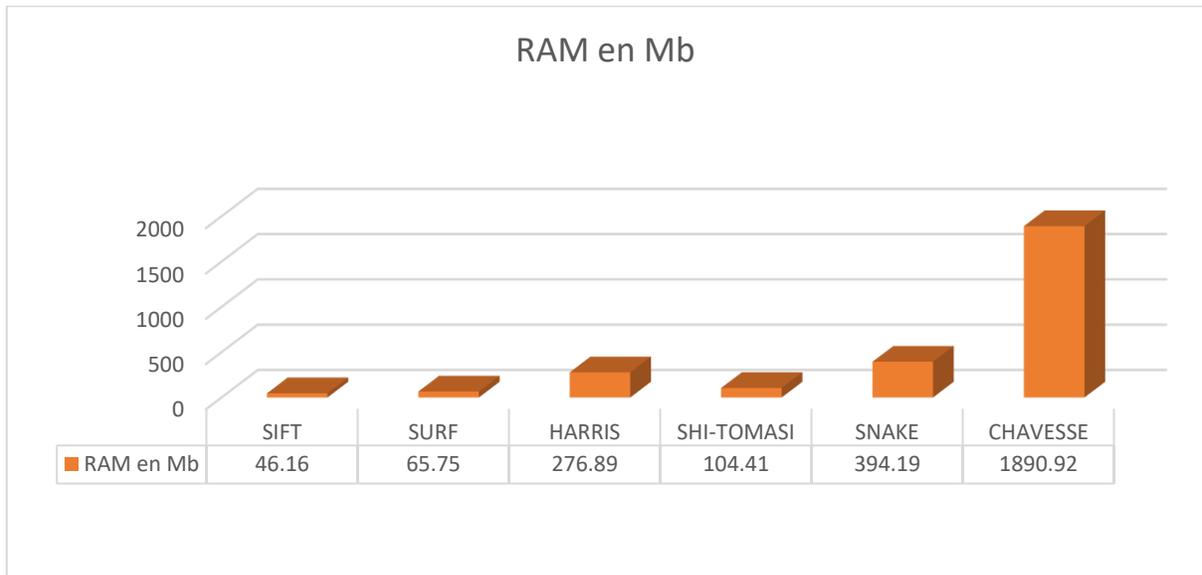
Las figuras 9 y 10 muestran el costo computacional en cuanto al consumo de tiempo de procesamiento en milisegundos (figura 9) y el consumo de memoria en MB (figura 10) para la obtención de los resultados de los algoritmos analizados.

Figura 9. Consumo de tiempo de procesamiento en milisegundos



Fuente: Elaboración propia

Figura 10. Consumo de memoria en MB



Fuente: Elaboración propia

Discusión

En un artículo similar de Ahmad *et al.* (2021) que aborda el procedimiento para clasificar y detectar la soya, se calcula el tiempo de ejecución de dicha actividad, y se puede observar que estos tiempos son ligeramente superiores a los presentados en el presente estudio.

Asimismo, en otro estudio realizado por Razfar *et al.* (2022), se propone un método de reconocimiento de la soya, para lo cual se mide tanto el consumo de memoria RAM como el tiempo necesario para llevar a cabo estas tareas. En síntesis, se evidencian tiempos similares a los presentados en esta investigación, aunque el uso de memoria es notablemente superior.

Como se ha observado durante la experimentación, existen diversos algoritmos que proporcionan diferentes características de las imágenes según las necesidades planteadas. En tal sentido, todos ellos tienen ventajas y desventajas dependiendo de los resultados que se busquen destacar.

Este trabajo de investigación tuvo como objetivo presentar algunos de esos algoritmos disponibles para el reconocimiento de imágenes, por lo que se puede indicar que la elección uno u otro dependerá de las necesidades específicas y del contexto donde se apliquen.

Finalmente, en lo que respecta al costo computacional, de acuerdo con los resultados obtenidos en cuanto al consumo de tiempo y memoria RAM, el algoritmo más destacado en

ambos aspectos fue el Shi-Tomasi, de ahí que podría ser la mejor elección para el reconocimiento de hojas de soya.

Conclusiones

En el presente trabajo se explicó el uso de diferentes tipos de algoritmos para el reconocimiento de hojas de soya. Esto se consiguió después de una exhaustiva investigación del estado de la cuestión con la finalidad de probar solo aquellos algoritmos reconocidos y recomendados por la literatura científica.

En este sentido, cabe indicar que se analizaron, programaron y se sometieron a prueba dichos algoritmos con diferentes imágenes para medir su costo computacional, lo cual arrojó resultados dispares. Aun así, se puede señalar que el algoritmo Shi-Tomasi puede ser elegido para reconocer hojas de soya, pues presenta un costo computacional aceptable para la tarea encomendada.

Futuras líneas de investigación

Como líneas de investigación se podrían agregar y probar más algoritmos de reconocimiento de imágenes, así como otras técnicas computacionales para detección de hojas de soya. Asimismo, se pueden probar los algoritmos que presenten mejor rendimiento extrapolar para reconocer no solo hojas de soya, sino otros tipos de plantas (p. ej., maíz, frijol, etc.).

Agradecimientos

Agradecemos al Dr. Miguel García Silvente, del Departamento de Ciencias de la Computación e Inteligencia Artificial, de la Universidad de Granada (España) por su amable dirección en la revisión bibliográfica y experimentación.

Referencias

- Ahmad, A., Saraswat, D., Aggarwal, V., Etienne, A. and Hancock, B. (2021). Performance of deep learning models for classifying and detecting common weeds in corn and soybean production systems. *Computers and Electronics in Agriculture*, 184, 106081. <https://doi.org/10.1016/j.compag.2021.106081>
- Alegre, E. y Fernández-Robles, L. (2016). SIFT (Scale Invariant Feature Transform). En G. Pajares, A. De la Escalera y E. Alegre (coord.), *Conceptos y métodos en visión por computador*. Grupo de Visión del Comité Español de Automática (CEA). <https://intranet.ceautomatica.es/sites/default/files/upload/8/files/ConceptosyMetodo senVxC.pdf>
- Aracil-López, R. (2012): *Desarrollo de un sistema cognitivo de visión para la navegación robótica* (trabajo de grado). Universidad Politécnica de Valencia. <https://riunet.upv.es/handle/10251/17010>
- Bay, H., Tuytelaars, T. and Luc Van Gool. (2006). SURF: Speeded Up Robust Features. *European Conference on Computer Vision* (pp. 404–417). https://doi.org/10.1007/11744023_32
- Cáceres Flórez, C. A., Ramos Sandoval, O. L. y Amaya Hurtado, D. (2015). Procesamiento de imágenes para reconocimiento de daños causados por plagas en el cultivo de *Begonia semperflorens* (flor de azúcar). *Acta Agronómica*, 64(3), 273–279. <https://doi.org/10.15446/acag.v64n3.42657>
- Canty, M. J. (2019). *Image analysis, classification and change detection in remote sensing: with algorithms for Python*. CRC Press.
- Enebral-González, J. (2009): *Detección y asociación automática de puntos característicos paradiferentes aplicaciones* (trabajo de grado). Universidad Politécnica de Cataluña. <https://upcommons.upc.edu/handle/2099.1/8052?locale-attribute=es>
- Fernández-García, N. (2012). *Introducción a la visión artificial* (trabajo de grado). Universidad de Córdoba. <http://www.uco.es/users/malfegan/2011-012/vision/Temas/Vision-artificial.pdf>
- Gómez-Fuentes, M. del C. y Cervantes-Ojeda, J. (2014). *Introducción al análisis y al diseño de algoritmos* (vol. 1). Universidad Autónoma Metropolitana. http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_Analisis_AlgoritmosVF.pdf

- Hernández-Juárez, S., Mejía-Rodríguez, A. R. y Arce-Santana, E. R. (2017). Dinámica pulmonar de estructuras anatómicas de interés en imágenes 4DCT. *Revista Mexicana de Ingeniería Biomédica*, 38(1), 126–140. <https://doi.org/10.17488/rmib.38.1.9>
- Jiménez-Carretero, D., Fernández-de-Manuel, L., Pascau, J., Tellado, J.M., Ramon, E., Desco, M., Santos, A. y Ledesma-Carbayo, M. J. *Algoritmo Level-Set para segmentación hepática en TAC con restricciones de curvatura local*. Actas del XXIX Congreso Anual de la Soc. Esp. Ing. Biomédica (pp. 207-210). Cáceres. Nov. 2011.
- Larese, M. G., Namias R. y Granitto P. M. (2011). *Análisis de imágenes para el estudio de la soja*. XIII Workshop de Investigadores en Ciencias de la Computación. http://sedici.unlp.edu.ar/bitstream/handle/10915/20051/Documento_completo.pdf?sequence=1
- Le, J. (12 de abril de 2018). *The 5 Computer Vision Techniques That Will Change How You See the World*. Medium. <https://heartbeat.comet.ml/the-5-computer-vision-techniques-that-will-change-how-you-see-the-world-1ee19334354b>
- Lin, Z. and Zhang, H. (2017). *Low-Rank Models in Visual Analysis*. Academic Press.
- Magallanes Estala, A., Díaz Franco, A., Reyes Rosas, M. A., Rosales Robles, E., Alvarado Carrillo, M., Silva Serna, M. M., Bustamante Dávila, A. J. y Cortinas Escobar, H. M. (2014). *Tecnología de producción en soja [Glycine max (L.) merrill] para el norte de Tamaulipas*. Instituto Nacional de Investigaciones Forestales, Agrícolas y Pecuarias. <http://www.inifapcirne.gob.mx/Biblioteca/Publicaciones/991.pdf>
- Mallick, S. (2016). *Image Recognition and Object Detection: Part 1 | Learn OpenCV*. <https://www.learnopencv.com/image-recognition-and-object-detection-part1/>
- Ñauñay-Ilbay, M. y Tipantuña-Córdova, L. (2013). *Análisis de eficiencia en algoritmos de reconocimiento de imágenes digitales aplicables a dispositivos móviles bajo la plataforma Android* (trabajo de grado). Universidad de las Fuerzas Armadas. <https://repositorio.espe.edu.ec/bitstream/21000/7774/1/T-ESPE-047430.pdf>
- Peña-Peñate, A., Silva-Rojas, L. G. y Alcolea-Núñez, R. (2016). Módulo de filtrado y segmentación de imágenes médicas digitales para el proyecto Vismedic. *Revista Cubana de Ciencias Informáticas*, 10(1), 13-27.
- Ramírez González, D., Pulido Sarmiento, G., Gerardino Arévalo, B., Cruz Romero, J., Estupiñán Escalante, E. y Cancino Suárez, S. (2009). Adquisición y reconocimiento

- de imágenes por medio de técnicas de visión e inteligencia artificial. *Iteckne*, 6(1), 5-13. <https://doi.org/https://doi.org/10.15332/iteckne.v6i1.290>
- Razfar, N., True, J., Bassiouny, R., Venkatesh, V. and Kashef, R. (2022). Weed detection in soybean crops using custom lightweight deep learning models. *Journal of Agriculture and Food Research*, 8, 100308. <https://doi.org/10.1016/j.jafr.2022.100308>
- Real Academia de Ingeniería (s. f.). Definición de *Costo computacional*. <http://diccionario.raing.es/es/lema/coste-computacional>
- Real Academia Española (s. f.). Definición de *Imagen*. <http://dle.rae.es/?id=KzwdY4y>
- Rivera-Díez, M. (2015). *Búsqueda de patrones para la mejora del proceso productivo y análisis de posicionamiento y profundidad* (trabajo de grado). Universidad de Sevilla. <https://idus.us.es/xmlui/handle/11441/40541>
- Rodríguez-Rieiro, B. (2011). *Segmentación estéreo mediante contornos activos* (tesis de maestría). Universidad de Sevilla. <https://biblus.us.es/bibing/proyectos/abreproy/70265>
- Roján-Herrera, O., Maqueira-López, L. A., Santana-Ges, I. de los M., Miranda-Sierra, C. A. y Núñez-Vázquez, M. (2022). Productividad de cultivares de soya en dos épocas de siembra. *Cultivos Tropicales*, 43(1). <https://doi.org/10.1234/ct.v43i1.1641>
- Rosas, J. C. y Young, R. (1991). *El cultivo de la soya*. Zamorano. <https://bdigital.zamorano.edu/server/api/core/bitstreams/028d6db0-2619-445e-a0e3-3befd9057b84/content>
- Sánchez-Prieto, I. (2014). *Procesado de imágenes médicas en MATLAB* (trabajo de grado). Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación. http://oa.upm.es/33072/1/TFG_irene_sanchez_prieto.pdf
- Trefethen, L. N. y Bau, D. (1997). *Numerical linear algebra*. Society For Industrial And Applied Mathematics.

| Rol de Contribución | Autor (es) |
|---|------------------------------------|
| Conceptualización | Felipe Anastacio González González |
| Metodología | Angel Mario Lerma Sánchez |
| Software | Joaquín Torres Mata |
| Validación | Irma Carolina González Sánchez |
| Análisis Formal | Felipe Anastacio González González |
| Investigación | Angel Mario Lerma Sánchez |
| Recursos | Joaquín Torres Mata |
| Curación de datos | Irma Carolina González Sánchez |
| Escritura - Preparación del borrador original | Felipe Anastacio González González |
| Escritura - Revisión y edición | Felipe Anastacio González González |
| Visualización | Felipe Anastacio González González |
| Supervisión | Felipe Anastacio González González |
| Administración de Proyectos | Angel Mario Lerma Sánchez |
| Adquisición de fondos | Joaquín Torres Mata |