

# Un algoritmo no supervisado aplicado a la segmentación de imágenes

Berns, Daniel Walther<sup>1</sup>

UNPSJB, Facultad de Ingeniería

<https://www.orcid.org/0000-0002-7231-6392>

[dwbarns@ing.unp.edu.ar](mailto:dwbarns@ing.unp.edu.ar)

## Resumen

En este trabajo mostramos un algoritmo no supervisado para segmentar imágenes, basado en la clasificación de píxeles mediante una red neuronal arbitraria cuyos coeficientes pueden ser calculados mediante la descomposición en valores singulares de una matriz. Proveemos un demo publicado en huggingface, y la teoría respectiva.

**Palabras claves:** Segmentación, Imagen, Algoritmo.

## A non supervised algorithm for images segmentation

### Abstract

In this work we show an unsupervised algorithm to segment images, based on the classification of pixels by means of an arbitrary neural network whose coefficients can be calculated by decomposing a matrix into singular values. We provide a demo posted on huggingface, and the respective theory.

**Keywords:** Segmentation, Image, Algorithm.

## 1. Introducción

La segmentación de una imagen digital (ver (*Papers With Code / Image Segmentation*, 2022)) es su división en partes con características distintivas determinadas. Esta tarea tiene múltiples aplicaciones en medicina (ver (Ronneberger y cols., 2015), (Ramesh y cols., 2021), (Liu y cols., 2021)), sensado remoto (ver (Kotaridis y Lazaridou, 2021) y (Jurado y cols., 2022)), control de tráfico vehicular (ver (L. Li y cols., 2018)), conteo de instancias (ver (Ren y Zemel, 2017) y (Waqas Zamir y cols., 2019)), y navegación de vehículos autónomos (ver (Trembl y cols., 2016) y (Papadeas y cols., 2021)).

<sup>1</sup> Profesor asociado de "Señales y Sistemas" de Ingeniería Electrónica, Facultad de Ingeniería, UNPSJB, Argentina. 23

Además, es posible obtener diferentes tipos de resultado. En la tarea de segmentación semántica se diferencian clases de objetos (personas, autos, edificios, zonas geográficas, condiciones climáticas, etc), mediante la clasificación de píxeles de acuerdo a categorías y regiones de la imagen, pero no se diferencian instancias de objetos (ver el trabajo sobre imágenes médicas de (Ronneberger y cols., 2015) y el modelo de (Wang y cols., 2022)). En la tarea de segmentación por instancias (ver Wei y cols. (2022) y (Ke y cols., 2021)), se busca distinguir distintos objetos dentro de una misma categoría. En la segmentación panóptica (ver (Kirillov y cols., 2019) y (Xiong y cols., 2019)), es posible distinguir entre diferentes clases e instancias y se unifican las tareas de segmentación semántica y de instancias.

En orden cronológico, tenemos diferentes algoritmos de segmentación: por agrupamiento (ver (Coleman y Andrews, 1979)), por "divisoria de aguas"(ver (Beucher, 1992)), por umbral (ver (Cheriet y cols., 1998)), por crecimiento de regiones (ver (Shih y Cheng, 2005)), por detección de bordes (ver (Muthukrishnan y Radha, 2011)), y con redes neurales profundas (deep learning neural networks) ((Badrinarayanan y cols., 2017), (Minaee y cols., 2021)).

Los algoritmos en uso actualmente, basados en redes neurales profundas, pueden consultarse en el sitio [paperswithcode.com](https://paperswithcode.com), en las páginas *Papers with code | semantic segmentation* (2022), *Papers with code | instance segmentation* (2022) y *Papers with code | panoptic segmentation* (2022). Si bien las publicaciones en cuestión obtienen resultados notables por su exactitud, las implementaciones disponibles tienen una gran complejidad tanto en el tamaño de los datos necesarios (modelos de redes neurales) como en la teoría necesaria para entender su funcionamiento. Por el contrario, en este trabajo consideramos un algoritmo implementado en menos de 70 líneas de python (empleando la librería `numpy`), basado en operaciones algebraicas básicas, y cuyos resultados, adecuados en la gran mayoría de las veces, sirven de línea de base para la verificación de otros métodos.

El algoritmo presentado aquí se basa en la clasificación de puntos en  $\mathfrak{R}^n$ , definida como una función  $f : \mathfrak{R}^n \times \{v : v \in N, p \in N, 2 \leq v < p\}$ .

## 2. Desarrollo

Consideremos las siguientes definiciones y convenciones:

1. Una clasificación es una función que operando sobre los elementos de un conjunto  $X$ , les asigna a cada uno de ellos un número o rótulo de clase  $c \in N : 0 \leq c < C$ , donde  $C + 1$  es el número total de clases (contando la clase con el rótulo 0).
2. Simbolizamos los vectores columnas con letras minúsculas en negrita  $\mathbf{v}$ , y los vectores fila o transpuestos  $\mathbf{v}^t$ .
3. Simbolizamos las matrices con letras mayúsculas en negrita  $\mathbf{Z}$ , y las matrices transpuestas  $\mathbf{Z}^t$ .

4. Dada una matriz  $\mathbf{P}$  con  $n$  filas y  $m$  columnas, simbolizamos sus elementos  $P_{i,j}$  (donde  $0 \leq i < n, 0 \leq j < m$ ), y sus columnas  $\mathbf{P}_j$ . (Esta es la convención del índice inicial 0 para arreglos numéricos).
5. Denotamos las matrices unitarias  $\mathbf{I}_r$  donde  $r$  indica la cantidad de 1 en su diagonal.
6. El símbolo  $\mathbf{u}_n$  es el vector columna de  $n$  elementos iguales a 1.

## 2.1. Un método de clasificación de elementos de $\mathfrak{R}^n$

Supongamos que disponemos de  $m$  datos  $\mathbf{X}_j \in \mathfrak{R}^n$ , y que construimos con ellos una matriz  $\mathbf{X} \in \mathfrak{R}^{n \times m}$ . La descomposición en valores singulares (ver (Strang, 2016), capítulo 7) es una operación que nos permite obtener los factores matriciales  $\mathbf{U} \in \mathfrak{R}^{n \times n}$ ,  $\mathbf{S} \in \mathfrak{R}^{n \times n}$ ,  $\mathbf{V}^t \in \mathfrak{R}^{n \times m}$  de la matriz

$$\mathbf{X} - \frac{\mathbf{X} \cdot \mathbf{u} \cdot \mathbf{u}^t}{m} = \mathbf{X} \cdot \left( \mathbf{I}_m - \frac{\mathbf{u} \cdot \mathbf{u}^t}{m} \right) = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^t, \quad (1)$$

donde  $\mathbf{u} \in \mathfrak{R}^m$ ,  $u_i = 1, 0 \leq i < m$ , e  $\mathbf{I}_m$  es la matriz unitaria  $\in \mathfrak{R}^{m \times m}$ . Además,  $\mathbf{U} \cdot \mathbf{U}^t = \mathbf{I}_n$ ,  $\mathbf{V}^t \cdot \mathbf{V} = \mathbf{I}_m$ , y la matriz  $\mathbf{S}$  es diagonal positiva. Observemos que el factor  $\left( \mathbf{I}_m - \frac{\mathbf{u}_m \cdot \mathbf{u}_m^t}{m} \right)$  tiene al menos un autovalor nulo, dado que

$$\left( \mathbf{I}_m - \frac{\mathbf{u}_m \cdot \mathbf{u}_m^t}{m} \right) \cdot \mathbf{u} = \mathbf{u} - \mathbf{u} \frac{\mathbf{u}^t \cdot \mathbf{u}}{m} = \mathbf{u} - \mathbf{u} \frac{m}{m} = \mathbf{0} \mathbf{u}. \quad (2)$$

Podemos construir una clasificación arbitraria de  $\mathbf{X}$  mediante los siguientes pasos

1. Transformación de  $\mathbf{X}$ : definimos

$$\mathbf{H} = \mathbf{S}_r^{-1} \cdot \mathbf{U}^t \cdot \mathbf{X} \cdot \left( \mathbf{I}_m - \frac{\mathbf{u} \cdot \mathbf{u}^t}{m} \right) = \mathbf{V}_r^t, \quad (3)$$

donde  $\mathbf{V}_r^t \in \mathfrak{R}^{r \times m}$  contiene las filas de  $\mathbf{V}_t$  correspondientes a los  $r$  autovalores más significativos de  $\mathbf{S}$ , y  $\mathbf{S}_r$  contiene los  $r$  autovalores más significativos de  $\mathbf{S}$ .

2. Normalización de  $\mathbf{H}$ : Definimos  $\mathbf{F}$  como una función lineal de  $\mathbf{H}$  de forma tal que

$$\delta \leq F_{i,j} \leq 1 - \delta, \quad 0 < \delta < \frac{1}{2}, \quad (4)$$

$$F_{i,j} = m H_{i,j} + b, \quad (5)$$

$$m = \frac{(1 - 2\delta)}{\max(\mathbf{H}) - \min(\mathbf{H})} \quad (6)$$

y

$$b = \frac{1 - \delta - m \max(\mathbf{H})}{m}. \quad (7)$$

3. Cuantización de **F**: dado un parámetro entero  $2 \leq \alpha$ , calculamos los dígitos en base  $\alpha$

$$G_{i,j} = \lfloor \alpha \times F_{i,j} \rfloor, \quad (8)$$

donde  $\lfloor x \rfloor = \max\{m \in \mathbb{Z} | m < x\}$ ,  $x \in \mathfrak{R}$  es la función parte entera de  $x$ .

4. Clasificación: finalmente calculamos los rótulos de las clases asociadas a cada  $\mathbf{X}_j$  como el producto

$$\mathbf{c} = \mathbf{w}^t \cdot \mathbf{G}, \quad (9)$$

donde  $w_i = \alpha^{n-1-i}$ , para  $0 \leq i < n$  y

$$0 \leq c_j = \sum_{i=0}^{n-1} \alpha^{n-1-i} G_{i,j} \leq \sum_{i=0}^{n-1} \alpha^{n-1-i} (\alpha - 1) = \alpha^{n-1} (\alpha - 1) \frac{1 - \frac{1}{\alpha^n}}{1 - \frac{1}{\alpha}} = \alpha^n - 1. \quad (10)$$

El parámetro  $\alpha$  determina la resolución de la clasificación (la cantidad de clases) como vemos en la ecuación 10. Si bien  $\alpha$  puede ser cualquier número entero mayor o igual a 2, el uso de números primos (3, 5, 7, 11, ...) evita que los límites de clases de resoluciones diferentes coincidan.

## 2.2. Segmentación de imágenes

Las imágenes digitales tomadas con cámaras comerciales son arreglos numéricos agrupados en tres o cuatro matrices, que denominamos **R**, **G**, **B**, y **A**, con números enteros positivos  $T = x : 0 \leq x \leq 255$ , de  $h$  filas y  $w$  columnas. Las matrices **R**, **G** y **B** contienen información sobre la intensidad de los colores Red (rojo), Green (verde) y Blue (azul) que componen un punto de color o pixel. Cuando está presente, la matriz **A** da información sobre la transparencia del pixel.

Definamos el pixel de la posición  $(k, l)$  en la imagen,  $0 \leq k < h$ ,  $0 \leq l < w$ , como la columna  $\mathbf{P}_j$ ,  $j = k \times w + l$

$$\mathbf{P}_j = \begin{bmatrix} R_{k,l} \\ G_{k,l} \\ B_{k,l} \end{bmatrix}, \quad (11)$$

de una matriz  $\mathbf{P} \in T^{3 \times (h \times w)}$  o

$$\mathbf{P}_j = \begin{bmatrix} R_{k,l} \\ G_{k,l} \\ B_{k,l} \\ A_{k,l} \end{bmatrix}, \quad (12)$$

de una matriz  $\mathbf{P} \in T^{4 \times (h \times w)}$  según dispongamos o no de la matriz **A**.

Aplicamos el algoritmo de clasificación a la matriz **P**, con una resolución  $2 \leq \alpha$  para obtener un

vector  $\mathbf{c}^t$  con las clases de cada columna. Dado que conociendo  $j$  y  $w$  podemos calcular

$$l = \text{mod}(j, h), \quad k = (j - l) // h, \quad (13)$$

donde  $\text{mod}$  es la función que nos da el resto de la división de números enteros  $//$ , es inmediato asignar la clase correspondiente a cada pixel de la imagen original. Así, podemos generar dos resultados diferentes

1. Mosaico: Una estructura de datos con una matriz en  $[0, 1]^{n \times m}$  por clase,

$$\begin{aligned} j &= k \times w + l, \\ c_j = p &\rightarrow \mathbf{K}_{k,l,p} = 1, \\ c_j \neq p &\rightarrow \mathbf{K}_{k,l,p} = 0. \end{aligned} \quad (14)$$

2. Imagen comprimida: los diferentes pixeles de cada clase son reemplazadas por un único pixel.

Este método de segmentación es una versión modificada de los propuestos en (Coleman y Andrews, 1979), basados en la determinación de grupos de pixeles mediante algoritmos de agrupamiento, según (Bell, 1966) y (Sinaga y Yang, 2020), y cuya complejidad en tiempo es NP Hard ((Mahajan y cols., 2009)). Dado que la complejidad temporal de los algoritmos de cálculo de descomposición en valores singulares es polinomial respecto al producto de las dimensiones de la matriz, cabe esperar una mejora en los tiempos de procesamiento (ver (X. Li y cols., 2019)).

### 2.3. Implementación

La implementación y una demostración en línea del algoritmo de segmentación está disponible en (Berns, 2022). Los programas están escritos en python, usando la librería `numpy` para el cálculo de la descomposición de valores singulares de una matriz, y la librería `gradio` para la interfase web. Como se puede ver en el sitio, se puede elegir el parámetro  $\alpha$ , y el tipo de resultado (vista en mosaico, o imagen simplificada). Se ha fijado el valor  $\delta = 0,001$ . En la solapa con el rótulo "Files" podemos ver todos los archivos de python con la implementación de la propuesta. Si bien la aplicación web tal como está publicada en [https://huggingface.co/spaces/DWB1962/svd\\_codes](https://huggingface.co/spaces/DWB1962/svd_codes) puede generar imágenes comprimidas, es preferible emplear el archivo `compress.py` ejecutado en forma local para una comparación justa, donde los archivos tienen el mismo formato y la misma cantidad de pixeles.

A continuación vemos las imágenes 1, 2 y 3, con los valores de parámetros  $\alpha = 13$  y  $\delta = 0,001$  (fijo en el programa).



Figura 1: Imagen original, tamaño 1654493 bytes.

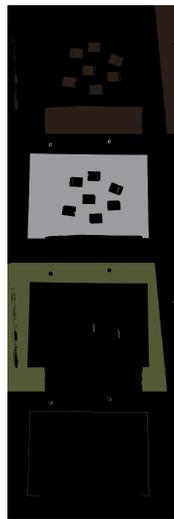


Figura 2: Mosaico,  $\alpha = 13$ .

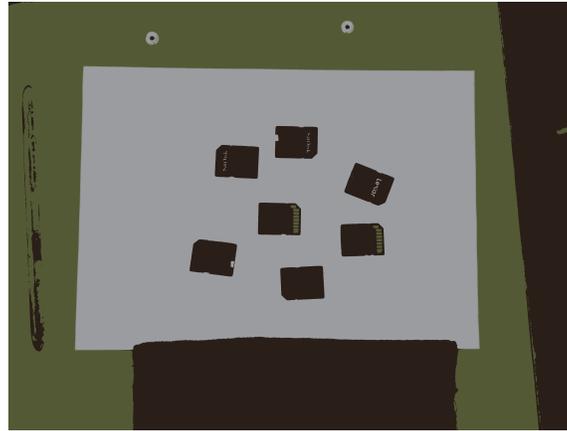


Figura 3: Imagen comprimida,  $\alpha = 13$ , tamaño 91498 bytes.

### 3. Conclusiones

Hemos presentado un algoritmo para segmentación de imágenes, basado en una clasificación arbitraria de píxeles determinada por la descomposición de valores singulares de una matriz. En base a los resultados obtenidos, es posible comprimir significativamente una imagen en la mayoría de los casos. Se ha publicado la implementación junto con una demostración en línea en el sitio *huggingface* (ver (Berns, 2022)). Existen algoritmos con mejor desempeño, basados en redes neurales, cuyos artículos e implementaciones pueden consultarse en las páginas *Papers with code | semantic segmentation* (2022), *Papers with code | instance segmentation* (2022) y *Papers with code | panoptic segmentation* (2022). Sin embargo, creemos que este trabajo muestra la posibilidad de obtener resultados alternativos con programas más sencillos (en cantidad de líneas y librerías usadas).

### Referencias

- Badrinarayanan, V., Kendall, A., y Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12), 2481–2495.
- Bell, G. H. (1966). *A comparison of some cluster-seeking techniques*. (Inf. Téc.). STANFORD RESEARCH INST MENLO PARK CALIF.
- Berns, D. W. (2022). *Huggingface | svdcodes*. <https://bit.ly/3gMAW07>. ((último acceso en 2022-10-25))

- Beucher, S. (1992). The watershed transformation applied to image segmentation. *Scanning Microscopy*, 1992(6), 28.
- Cheriet, M., Said, J. N., y Suen, C. Y. (1998). A recursive thresholding technique for image segmentation. *IEEE transactions on image processing*, 7(6), 918–921.
- Coleman, G. B., y Andrews, H. C. (1979). Image segmentation by clustering. *Proceedings of the IEEE*, 67(5), 773–785.
- Jurado, J. M., López, A., Pádua, L., y Sousa, J. J. (2022). Remote sensing image fusion on 3d scenarios: A review of applications for agriculture and forestry. *International Journal of Applied Earth Observation and Geoinformation*, 112, 102856. Descargado de <https://www.sciencedirect.com/science/article/pii/S1569843222000589> doi: <https://doi.org/10.1016/j.jag.2022.102856>
- Ke, L., Tai, Y.-W., y Tang, C.-K. (2021). Deep occlusion-aware instance segmentation with overlapping bilayers. En *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4019–4028).
- Kirillov, A., He, K., Girshick, R., Rother, C., y Dollár, P. (2019). Panoptic segmentation. En *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9404–9413).
- Kotaridis, I., y Lazaridou, M. (2021). Remote sensing image segmentation advances: A meta-analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173, 309–322.
- Li, L., Qian, B., Lian, J., Zheng, W., y Zhou, Y. (2018). Traffic scene segmentation based on rgb-d image and deep learning. *IEEE Transactions on Intelligent Transportation Systems*, 19(5), 1664-1669. doi: 10.1109/TITS.2017.2724138
- Li, X., Wang, S., y Cai, Y. (2019). Tutorial: Complexity analysis of singular value decomposition and its variants. *arXiv preprint arXiv:1906.12085*.
- Liu, X., Song, L., Liu, S., y Zhang, Y. (2021). A review of deep-learning-based medical image segmentation methods. *Sustainability*, 13(3), 1224.
- Mahajan, M., Nimbhorkar, P., y Varadarajan, K. (2009). The planar k-means problem is np-hard. En *International workshop on algorithms and computation* (pp. 274–285).
- Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., y Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*.

- Muthukrishnan, R., y Radha, M. (2011). Edge detection techniques for image segmentation. *International Journal of Computer Science & Information Technology*, 3(6), 259.
- Papadeas, I., Tsochatzidis, L., Amanatiadis, A., y Pratikakis, I. (2021). Real-time semantic image segmentation with deep learning for autonomous driving: A survey. *Applied Sciences*, 11(19), 8802.
- Papers with code | image segmentation.* (2022). <https://bit.ly/3sBdNkp>. ((último acceso en 2022-10-25))
- Papers with code | instance segmentation.* (2022). <https://bit.ly/3TXr1Dx>. ((último acceso en 2022-10-25))
- Papers with code | panoptic segmentation.* (2022). <https://bit.ly/3SUayzn>. ((último acceso en 2022-10-25))
- Papers with code | semantic segmentation.* (2022). <https://bit.ly/3N91ho4>. ((último acceso en 2022-10-25))
- Ramesh, K., Kumar, G. K., Swapna, K., Datta, D., y Rajest, S. S. (2021). A review of medical image segmentation algorithms. *EAI Endorsed Transactions on Pervasive Health and Technology*, 7(27), e6–e6.
- Ren, M., y Zemel, R. S. (2017, July). End-to-end instance segmentation with recurrent attention. En *Proceedings of the IEEE conference on computer vision and pattern recognition (cvpr)*.
- Ronneberger, O., Fischer, P., y Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. En *International conference on medical image computing and computer-assisted intervention* (pp. 234–241).
- Shih, F. Y., y Cheng, S. (2005). Automatic seeded region growing for color image segmentation. *Image and vision computing*, 23(10), 877–886.
- Sinaga, K. P., y Yang, M.-S. (2020). Unsupervised k-means clustering algorithm. *IEEE access*, 8, 80716–80727.
- Strang, G. (2016). *Introduction to linear algebra*. Wellesley-Cambridge Press. Descargado de <https://books.google.com.ar/books?id=efbxjwEACAAJ>
- Treml, M., Arjona-Medina, J., Unterthiner, T., Durgesh, R., Friedmann, F., Schuberth, P., ... others (2016). Speeding up semantic segmentation for autonomous driving.
- Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., ... others (2022). Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*.

## Revista REDIUNP

Revista Electrónica de divulgación de Metodologías Emergentes  
en el Desarrollo de STEM  
Aprobada en Consejo Directivo de la Facultad RCDFI-419-2018  
ISSN:2683-8648  
Vol. 4 N° 2(2022)



Berns, Daniel Walther  
*Un algoritmo no supervisado aplicado a la segmentación de imágenes.*

Waqas Zamir, S., Arora, A., Gupta, A., Khan, S., Sun, G., Shahbaz Khan, F., . . . Bai, X. (2019). isaid: A large-scale dataset for instance segmentation in aerial images. En *Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops* (pp. 28–37).

Wei, Y., Hu, H., Xie, Z., Zhang, Z., Cao, Y., Bao, J., . . . Guo, B. (2022). Contrastive learning rivals masked image modeling in fine-tuning via feature distillation. *arXiv preprint arXiv:2205.14141*.

Xiong, Y., Liao, R., Zhao, H., Hu, R., Bai, M., Yumer, E., y Urtasun, R. (2019). Upsnet: A unified panoptic segmentation network. En *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8818–8826).