

# Método integral de segmentación automática de cromosomas basado en redes neuronales profundas

Sebastián Fenoglio<sup>1</sup>, César Martínez<sup>1,2</sup>, and Matías Gerard<sup>1\*</sup>

<sup>1</sup> Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i)  
Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral - CONICET

<sup>2</sup> Laboratorio de Cibernética, Facultad de Ingeniería, Universidad Nacional de Entre Ríos  
(\* ) mgerard@sinc.unl.edu.ar  
<http://sinc.unl.edu.ar>

**Resumen** En este trabajo se aborda el problema de la segmentación automática de cromosomas con tinción en banda G mediante la aplicación de redes convolucionales profundas. Se propone una metodología para generación de datos sintéticos y se exploran diferentes arquitecturas. Los resultados muestran desempeños mayores a 93% en recall y cercanos a 90% con el coeficiente de Jaccard. Además, se aplican métodos de post-procesamiento para mejorar la probabilidad de que la red convolucional acierte en el número de cromosomas presentes en un solapamiento.

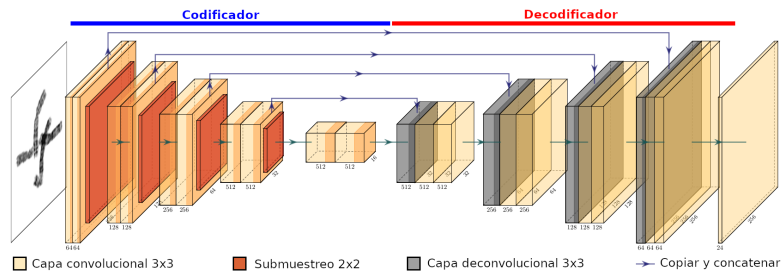
**Palabras clave:** segmentación de cromosomas, segmentación de imágenes, redes convolucionales, aprendizaje profundo.

## 1. Introducción

La citogenética es el campo de la genética que estudia la estructura, función y comportamiento de los cromosomas. Una representación frecuentemente utilizada en el análisis de los cromosomas es el cariograma [1]. Éste consiste en una representación ordenada de los 23 pares de cromosomas, en seres humanos, a partir de una imagen fotográfica. Un paso fundamental en la obtención del cariograma es la segmentación de los cromosomas a partir de dicha imagen.

Un problema encontrado es el solapamiento de cromosomas y las ambigüedades que existen sobre cómo separarlos. Éste ha sido abordado empleando principalmente métodos geométricos basados en el análisis de contornos. Un método frecuente es la búsqueda de puntos de menor concavidad para trazar una línea de separación [1, 2], otros métodos se basan en la triangulación Delaunay [3] de los píxeles del contorno y en el seguimiento de los píxeles de mayor intensidad [4]. Asimismo, estos enfoques abordan el solapamiento de 2 cromosomas y sólo en [2] se propone la construcción de un árbol en el que se aplican las operaciones y se analizan las hojas para determinar la combinación de operaciones correcta.

Las redes neuronales convolucionales (CNN en inglés) son modelos ampliamente empleados en visión computacional. Dos de las arquitecturas más conocidas son U-Net [5] y VGG [6]. Éstos modelos han demostrado un gran desempeño



**Figura 1:** Arquitectura neuronal propuesta.

en tareas de segmentación [7] y clasificación de imágenes [6]. Sin embargo, su uso en la segmentación de cromosomas ha sido poco explorado, siendo Hu et al. [8] los únicos que las han aplicado para esta tarea. Su propuesta consiste en una versión simplificada de U-Net, y fue empleada para separar casos de dos cromosomas solapados empleando la tinción M-FISH, obteniendo buenos resultados.

En este trabajo se propone un nuevo modelo inspirado en U-Net para la segmentación automática de cromosomas en metafase empleando tinción en banda G y se compara el desempeño frente a otras arquitecturas. También se explora el uso de diferentes esquemas de post-procesamiento para mejorar la calidad de las segmentaciones producidas.

## 2. Modelo neuronal y post-procesamiento

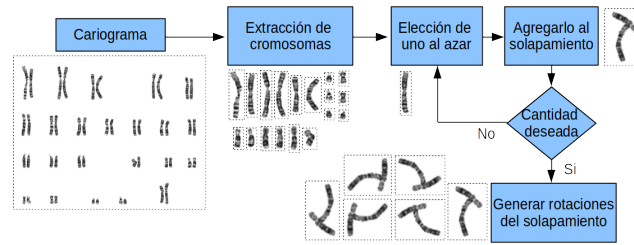
### 2.1. Arquitectura neuronal

La arquitectura elegida Hu4N es una variación de la propuesta por Hu [8] e inspirada en U-net [5], que puede observarse en la Figura 1. La imagen de entrada es normalizada con una media de 0.5 y un desvío de 0.5. La capa de salida consta de 24 canales por cada píxel debido a las 24 categorías consideradas: el fondo y los 23 tipos de cromosomas. Estas 23 clases surgen de los 22 pares de cromosomas homólogos y de otra adicional que agrupa al par sexual X e Y. Cabe recalcar que con esto no sólo permitiría separar los cromosomas, sino que también clasificarlos. La lógica tras la arquitectura es combinar información local u obtenida de las primeras capas del codificador con información global obtenida al final del mismo.

### 2.2. Métodos de post-procesamiento

Para la etapa de post-procesamiento de los resultados se proponen 3 métodos: Red W, Corrección por canales, y Eliminación de pequeñas imperfecciones. Estos pueden ser aplicados independientemente o en cascada. A continuación se describe el funcionamiento de cada método.

**Red W.** Consiste en acoplar, a la salida de la red Hu4N utilizada para la separación de solapamientos, otra instancia de la misma red que toma como



**Figura 2:** Diagrama de bloques del proceso de generación de datos sintéticos.

entrada la salida de los 24 canales de la primera. El conjunto se llamará Red W y se referirá como primer y segunda U a cada instancia de la red Hu4N. La lógica es que la segunda U aprenda a corregir los errores de la primera.

**Corrección por canales (CC).** Se definen las clases confiables como aquellas para las que al menos un píxel posee una probabilidad de pertenencia a dicha clase mayor que un umbral. Luego, las clases con probabilidades inferiores al umbral son descartadas y se recalculan las probabilidades para las clases restantes. Con un umbral suficientemente elevado se puede lograr reducir el error de aparición de clases incorrectas.

**Eliminación de pequeñas imperfecciones (EPI).** Consiste en rectificar pequeñas zonas de píxeles de una misma clase que no tienen el tamaño suficiente para ser un cromosoma. Como estas imperfecciones pueden ser consecuencia de errores de la predicción de la red o de la imagen de entrada, la propuesta es analizar la vecindad de las zonas de píxeles desechadas para asignarle una clase.

### 3. Datos y medidas de evaluación

El dataset fue generado a partir de la base de datos reportada en [9]. Ésta consiste en imágenes microscópicas de cromosomas con tinción en banda G para 612 células y sus correspondientes cariogramas. Con ellas, se generó el dataset utilizado considerando solapamientos de entre 2 y 5 cromosomas, verificando que todas las clases estén presentes de forma balanceada en el conjunto de datos.

La generación de datos se realizó siguiendo el esquema de la Figura 2. Dado un cariograma, se hace la extracción de los cromosomas individuales, la generación de solapamientos (dentro de porcentajes predefinidos) incluyendo tantos cromosomas como se especifique, y la posterior construcción de versiones rotadas del solapamiento generado. En este paso, a fin de agregar variabilidad a los datos, además se aplica *histogram matching* al grupo de píxeles no pertenecientes al fondo con una media de 128 y un desvío aleatorio en el rango [50, 70], y luego un filtro gaussiano de 3x3 con desvío al azar en el rango [0, 0.7].

El desempeño de las redes consideradas se evaluó mediante el recall y el coeficiente de Jaccard. El recall se calcula como:

$$recall = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}} \quad (1)$$

y mide la capacidad que tiene el modelo de distinguir los píxeles correctamente de cada clase. El coeficiente de Jaccard se calcula como:

$$Jaccard = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos} + \text{Falsos Positivos}} \quad (2)$$

y evalúa la similaridad entre la predicción y el ground truth, penalizando además por los píxeles incorrectamente clasificados. Para evaluar la capacidad para separar los solapamientos correctamente se definió una nueva medida denominada confiabilidad, que se calcula como:

$$conf = \frac{\text{Cantidad de predicciones acertadas en número de cromosomas}}{\text{Cantidad total de solapamientos}} \quad (3)$$

que indica la probabilidad de que el número de cromosomas obtenidos luego de la segmentación sea el correcto. Así, una confiabilidad de 95 % significa que en el 95 % de los casos la red convolucional acierta en la cantidad de cromosomas de un solapamiento.

## 4. Resultados

El dataset se particionó en 3 grupos: train, validation y test. El entrenamiento se llevó a cabo empleando mini-batch de entre 10 y 25 imágenes sobre los datos de train, mientras que validation fue empleado para analizar la capacidad de generalización y detener el entrenamiento. En todos los experimentos se utilizó el optimizador Adam [10] con tasa de aprendizaje  $10^{-4}$  y la función de pérdida de entropía cruzada. Los resultados presentados corresponden a la evaluación de las redes sobre el conjunto de test.

### 4.1. Determinación de la mejor arquitectura

Se compara la arquitectura propuesta con diferentes variantes que se tuvieron en cuenta:

- Hu2: arquitectura usada en [8], por tener 2 niveles de profundidad dados por la cantidad de capas de submuestreo del codificador e idéntica cantidad de capas de sobremuestreo del decodificador.
- Hu3: agrega un nivel de profundidad.
- Hu4 y Hu4N: con un nivel adicional, en cada caso agregando capas convolucionales de forma de mantener la simetría entre el codificador y el decodificador. La diferencia entre Hu4 y Hu4N es que la primera no realiza la normalización de entrada de los datos.
- VGG16: variación de la arquitectura VGG16 [6] en la que se modifica el decodificador para que sea convolucional y utilice todos los resultados obtenidos inmediatamente antes de cada submuestreo.

**Tabla 1:** Resultados en la búsqueda de la mejor arquitectura.

| Modelo  | Hu2     | VGG16   | Hu3     | Hu4     | Hu4N           |
|---------|---------|---------|---------|---------|----------------|
| Épocas  | 20      | 13      | 17      | 9       | 12             |
| Recall  | 72,48 % | 58,31 % | 84,16 % | 86,91 % | <b>88,71 %</b> |
| Jaccard | 65,58 % | 49,84 % | 78,87 % | 80,70 % | <b>83,47 %</b> |

**Tabla 2:** Resultados de la comparación de métodos de post-procesamiento.

| Método   | Hu4N         | Hu4N + CC <sup>90</sup><br>+ EPI <sup>100</sup> | Red W               | Red W + CC <sup>90</sup><br>+ EPI <sup>100</sup> |
|----------|--------------|---|---------------------|--|
| Recall   | 88,9 ± 0,6 % | 89,1 ± 0,6 %                                    | <b>93,3 ± 0,4 %</b> | <b>93,3 ± 0,5 %</b>                              |
| Jaccard  | 87,2 ± 0,6 % | 86,4 ± 0,7 %                                    | <b>92,0 ± 0,5 %</b> | 91,5 ± 0,5 %                                     |
| Confiab. | 59,9 ± 2,4 % | 81,9 ± 1,5 %                                    | 70,9 ± 2,0 %        | <b>90,0 ± 1,1 %</b>                              |

Los resultados obtenidos se presentan en la Tabla 1. Se observa que la red VGG16 tiene un menor desempeño en ambas medidas, mientras que la red Hu3 mejora más de un 10% tanto el recall como el Jaccard. Los mejores resultados en recall y Jaccard son obtenidos con Hu4N, con un aumento de 2% en ambas respecto a Hu4. Por otro lado, las redes Hu4 y Hu4N obtienen mejores resultados por tener una mayor cantidad de parámetros entrenables (20 millones) respecto a Hu2 (2 millones), Hu3 (8 millones) y VGG16 (14 millones). Esta última también puede justificar su mal desempeño en que fue entrenada con un dataset de otro problema y utilizada mediante transfer learning en este trabajo, por lo que probablemente las características que extraiga de las imágenes no sean las más adecuadas. Por último, comparando Hu4 y Hu4N se observa que la capa de normalización de las imágenes de entrada mejora el rendimiento de la red.

#### 4.2. Comparación de métodos de post-procesamiento

En experimentos preliminares se determinaron los parámetros de los métodos CC y EPI utilizando un grupo de 2688 imágenes. Se genera, además, un nuevo grupo de 2688 imágenes bajo las mismas condiciones que el anterior con el fin de conocer el desempeño de los métodos en datos no vistos anteriormente. Se presentan las medidas de recall, Jaccard y confiabilidad en la Tabla 2, indicando como superíndice los parámetros de cada método. Tanto en el recall como en Jaccard se ve una amplia mejoría del método Red W + CC<sup>90</sup> + EPI<sup>100</sup>, en ambos casos sacando una diferencia mayor de 4% a los demás. Dicho método también es superior en cuanto a la confiabilidad, siendo ampliamente mayor que los demás. Puede verse que la combinación CC<sup>90</sup> + EPI<sup>100</sup> prácticamente no modifica el recall y el Jaccard, pero sí mejora notablemente la confiabilidad. Se concluye que con la exploración de arquitecturas de redes convolucionales más complejas podrían mejorarse los resultados obtenidos.

## 5. Conclusiones

En este trabajo se propone un modelo neuronal basado en la red U para la segmentación automática de cromosomas. La comparación de ésta propuesta con otras arquitecturas indica que el modelo desarrollado presenta un mejor desempeño en términos de recall y Jaccard. Se logra un rendimiento aún mejor con el agregado de otra red U como etapa de post-procesamiento. Además, la incorporación de los métodos de corrección por canales y eliminación de imperfecciones mejoran de forma considerable los resultados en términos de confiabilidad.

En base a los resultados alcanzados, sería interesante evaluar otros modelos neuronales más sofisticados, tales como Mask R-CNN, que utiliza dos redes convolucionales: una especializada en la detección de objetos y otra que usa esa información junto a la imagen de entrada para clasificar dichos objetos.

### Agradecimientos

En este trabajo se utilizó el cluster Pirayu, Agencia Santafesina de Ciencia, Tecnología e Innovación (ASACTEI), Santa Fe, el cual forma parte del Sist. Nac. de Computación de Alto Desempeño.

### Referencias

- [1] S. Minaee, M. Fotouhi, and B. H. Khalaj, "A geometric approach to fully automatic chromosome segmentation," in *Signal Processing in Medicine and Biology Symposium (SPMB), 2014 IEEE*, pp. 1–6, IEEE, 2014.
- [2] E. Grisan, E. Poletti, and A. Ruggeri, "Automatic segmentation and disentangling of chromosomes in q-band prometaphase images," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 4, pp. 575–581, 2009.
- [3] W. Srisang, K. Jaroensutasinee, and M. Jaroensutasinee, "Segmentation of overlapping chromosome images using computational geometry," *Walailak Journal of Science and Technology (WJST)*, vol. 3, no. 2, pp. 181–194, 2011.
- [4] D. Somasundaram, S. Palaniswami, and V. Venkatesakumar, "G-band chromosome segmentation, overlapped chromosome separation and visible band calculation," *International Journal of Human Genetics*, vol. 14, no. 2, pp. 73–81, 2014.
- [5] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.
- [6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.
- [8] R. L. Hu, J. Karnowski, R. Fadely, and J.-P. Pommier, "Image segmentation to distinguish between overlapping human chromosomes," *arXiv preprint arXiv:1712.07639*, 2017.
- [9] G. Ritter and L. Gao, "Automatic segmentation of metaphase cells based on global context and variant analysis," *Pattern Recognition*, vol. 41, no. 1, pp. 38–55, 2008.
- [10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.