

## PROCESAMIENTO DIGITAL DE TEXTO MANUSCRITO PARA LA GENERACION DE CORPUS DE RECONOCIMIENTO AUTOMATICO DE ESCRITURA

Oswaldo Rodríguez Hernández<sup>1</sup> Manuel Ornelas Rodríguez<sup>2</sup>

### RESUMEN

El procesamiento digital de imágenes consiste en usar algoritmos computacionales y ordenadores digitales para convertir una imagen dada en una imagen digital y obtener información de ella. El proyecto que se presenta consiste en conformar una herramienta desarrollada en la plataforma Java con la cual se pueda pre-procesar imágenes de texto manuscrito y así generar un corpus de imágenes pre-procesadas para su posterior uso en el área de reconocimiento automático de escritura. Se implementan y describen algoritmos para la reducción de ruido en la imagen, selección de umbral óptimo y binarización de un conjunto de imágenes de texto manuscrito adquiridas de diferentes fuentes, así como distintas resoluciones, caligrafía, iluminación etc.

**PALABRAS CLAVE** Visión artificial, tratamiento digital de imágenes, reconocimiento de escritura, transcripción de texto.

---

<sup>1</sup> Instituto Tecnológico de León, Ingeniería en Sistemas Computacionales, Av. Tecnológico S/n, Fracc. Industrial Julián de Obregón, 37290 León, Guanajuato, (477) 710 5200.

<sup>2</sup> Profesor, Instituto Tecnológico de León, División de Estudios de Posgrado e Investigación, Av. Tecnológico S/n, Fracc. Industrial Julián de Obregón, 37290 León, Guanajuato, (477) 710 5200; mornelas67@yahoo.com.mx

## Introducción

Una imagen es una representación plana de un objeto real en 3 dimensiones, partiendo de esa premisa el procesamiento digital de imágenes consiste en usar algoritmos computacionales y ordenadores digitales para convertir una imagen dada en una imagen digital. Una imagen digital está compuesta por un número finito de elementos, con una localización y valor particular, estos elementos los denominamos píxeles. (Gonzales & Woods, 2008).

El procesamiento digital de imágenes y muchas de las técnicas utilizadas surgen a mediados de 1960 en el Instituto Tecnológico de Massachusetts, la Universidad de Maryland y algunas otras instituciones de investigación, pero principalmente en el Jet Propulsion Laboratory por el ingeniero de la NASA el Dr. Robert Nathan, el cual desarrolló el primer software operacional de procesamiento digital de imágenes, esto debido a que las imágenes recibidas de una sonda no tripulada enviada a la luna, resultaron distorsionadas.

Durante el procesamiento de la imagen de texto manuscrito se contemplan dos fases principales, la etapa de pre-procesamiento y la etapa de segmentación. (Rajput, Rajput, & Badave, 2012)

Los objetivos principales de la etapa de pre-procesamiento son:

Reducción de ruido.

Normalización de los datos.

En la etapa de pre-procesamiento se pretende minimizar todos aquellos elementos en la imagen que afectan el reconocimiento de la escritura, elementos como la iluminación no uniforme, ruido en la imagen, etc.

En la etapa de segmentación identifican 2 tipos de enfoques:

La segmentación orientada a bordes.

La segmentación orientada a regiones.

La etapa de segmentación contempla la separación de las líneas de texto en la imagen de texto manuscrito así como las palabras para poder facilitar la tasa de reconocimiento en fases posteriores.

## Métodos y materiales

Con el fin de poder aplicar y probar las diversas técnicas de procesamiento digital de imágenes así como su posterior extracción de características es necesario contar con una base de datos de imágenes sobre las cuales realizar dichas pruebas.

Para cumplir el objetivo, se recopilaron 49 imágenes de texto manuscrito de diversas fuentes con diferentes tipos de escritura, iluminación, fondo, etc. (Ver figura 1). Además de utilizar la base de datos IAM Handwriting Database para las pruebas de segmentación. (Liwicki & Bunke, 2002)

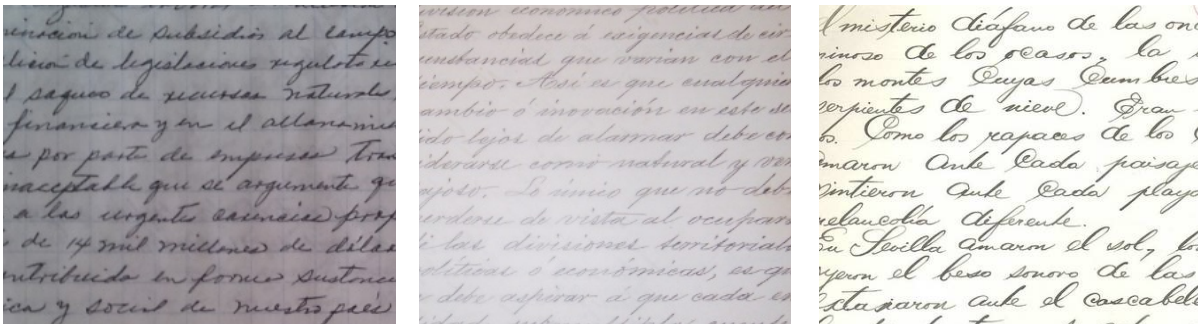


Figura 1.- Muestra de las diferentes imágenes recolectadas

Gran parte de las imágenes que conforman la base de datos fueron capturadas mediante una cámara digital, con diferentes condiciones de luz. Además las imágenes presentan un fondo cambiante por lo que fueron recortadas manualmente con el fin de extraer solo la parte de la imagen que contiene texto manuscrito.

Al recortar la imagen para extraer solo el texto se modifica su tamaño lo que provoca que todas las imágenes que conforman la base de datos tengan tamaños diferentes.

Para simplificar la manipulación de las imágenes es necesario reducir la cantidad de información en las mismas, es decir reducir el vector de colores en las imágenes. Para ello es necesario convertir la imagen a escala de grises y de esta forma la imagen tendrá como información solo una escala de intensidades con la que es más sencillo trabajar.

No existe una forma única de convertir una imagen en escala de grises, sin embargo por razones de practicidad utilizamos una de las formas más sencillas como se muestra en (1).

$$Gris = \frac{Rojo+Verde+Azul}{3} \quad (1)$$

Una vez que la imagen se convierte a escala de grises se continúa con el proceso de binarización el cual consiste en convertir una imagen en escala de grises a una imagen en blanco y negro (0 o 1). Para poder llevar a cabo la binarización es necesario seleccionar un valor de umbral; Para la selección del umbral óptimo existen múltiples métodos, para nuestro caso se codificaron y probaron 2 diferentes métodos: Otsu y Ridler-Calvard.

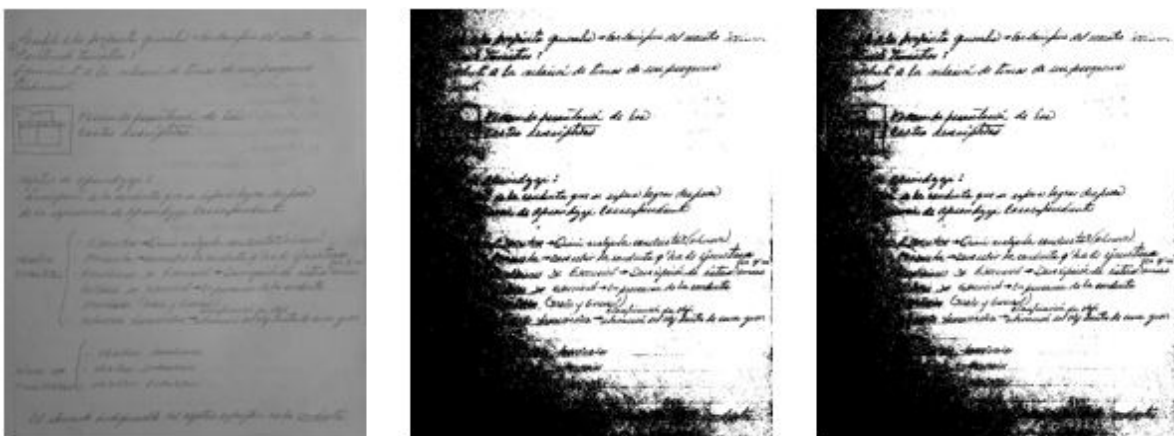
El método de Otsu calcula el umbral óptimo de forma que la dispersión dentro de cada clase, medida mediante la varianza, sea lo más pequeña posible, pero al mismo tiempo la dispersión entre clases diferentes sea lo más alta posible.

El método Ridler-Carvald, al igual que el método de Otsu trata de dividir la imagen en dos clases principales, el fondo y el objeto de interés, para implementarlo es necesario asignar un valor de umbral inicial, generalmente un valor intermedio ya que el umbral óptimo usualmente no está muy alejado de este valor. Con este umbral inicial arbitrario se dividen los pixeles de la imagen en dos clases: una por arriba del umbral y una por debajo (el fondo de la imagen y el objeto de interés). Se calcula la media de los valores de los pixeles de cada clase y el nuevo valor del umbral estará dado por la media de la media de las clases. Se repite el proceso hasta que la diferencia entre el umbral y su inmediato anterior convergen a un valor de error establecido.

## Resultados

Al implementar los algoritmos descritos en la sección anterior, se obtienen resultados similares y con pérdidas de texto notables.

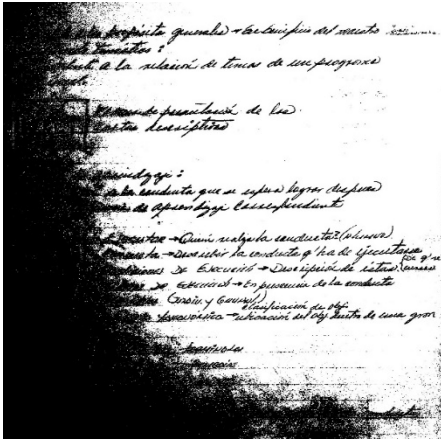
La figura 2 muestra los resultados de la binarización con los umbrales obtenidos por los métodos aplicados, para el caso de esta imagen el umbral óptimo con el método Otsu fue de 136 y con Ridler-Carvald se obtuvo un valor de 135.



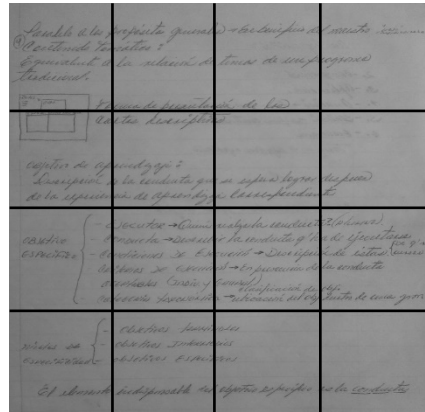
**Figura 2.- Umbral por métodos: Otsu y Ridler-Calvard. (a) Imagen de entrada en escala de grises. (b) imagen binarizada con umbral Otsu. (c) Imagen binarizada con umbral Ridler-Calvard.**

Al aplicar los métodos de binarización en algunas de las imágenes de texto manuscrito que conforman la base de datos, se presenta un inconveniente: no se realiza una correcta separación del fondo de la imagen y se genera una pérdida de información. La razón de esta pérdida es porque los métodos implementados calculan el umbral tomando en cuenta todos los píxeles que componen la imagen, por lo tanto las regiones en las que la iluminación no es óptima, es decir, aquellas regiones más oscuras de la imagen, afectan el valor final del umbral.

La figura 3 muestra una imagen binarizada con un umbral de 136 por el método de Otsu, en esta figura se aprecia cómo afecta la iluminación deficiente en el resultado final de la binarización



**Figura 3.- Imagen binarizada con valor de umbral de 136 por el método Otsu.**



**Figura 3.- Imagen subdividida.**

Para corregir el problema anterior, se subdividió la imagen principal en varias sub-imágenes, esto con el fin de aislar lo más posible las zonas oscuras de la misma. Una vez realizada la subdivisión de las imágenes se calculó el umbral óptimo para cada una de ellas. En el caso específico de la figura 4, la subdivisión que muestra mejores resultados es una de 16 sub-imágenes. Para cada una de estas se sub-imágenes se calculó un umbral con el método de Otsu, obteniendo los resultados mostrados en la tabla 1.

Tabla 1.- Resultados por el método Otsu para cada sub-imagen

Sub-imagen	Umbral	Sub-imagen	Umbral
1	118	9	114
2	129	10	125
3	135	11	129
4	138	12	125
5	115	13	107
6	129	14	114
7	135	15	118
8	137	16	120

Una vez que cada sub-imagen es binarizada con un valor de umbral independiente se vuelven a unir todas para conformar una sola imagen binarizada. Los resultados de dicha binarización se muestran en la figura 5. La binarización final se observa notablemente mejorada, se utilizan 16 diferentes umbrales en lugar de uno solo de 136.

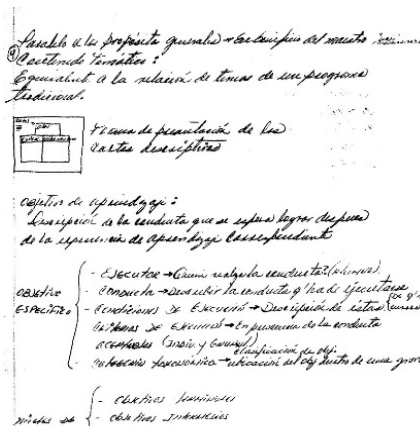


Figura 4.- Imagen Final de la binarización

## Conclusiones

En algunos casos donde no es posible digitalizar las imágenes apropiadamente, con una buena iluminación proporcionada por un escáner por ejemplo, es necesario buscar alternativas, las cuales presentan sus ventajas y desventajas.

Los resultados obtenidos en este trabajo muestran la factibilidad de que los algoritmos de binarización implementados obtienen mejores resultados en imágenes con iluminación deficiente cuando estas son divididas en sub-imágenes y tratadas de forma independiente.

Como trabajo futuro se realizara la separación de renglones en la imagen de texto manuscrito así como la corrección y la uniformidad de la escritura, todo esto para facilitar el trabajo posterior del reconocimiento automático. Además de implementar nuevos algoritmos de binarización para poder cotejar resultados no solo de dos métodos.

## REFERENCIAS

- Aly, A. A., Deris, S. B., & Zaki, N. (Octubre de 2011). Research Review for Digital Image Segmentation Techniques. *International Journal of Computer Science & Information Technology*, Vol. 3(No. 5). Recuperado el 5 de Noviembre de 2013, de <http://airccse.org/journal/jcsit/10111csit09.pdf>
- Gonzales, R. C., & Woods, R. E. (2008). *Digital Image Processing* (Tercera ed.). New Jersey: Pearson Prentice Hall.
- Kaseko, M. S., & Ritchie, S. G. (1993). A neural network-based methodology for pavement crack detection and classification. *Transportation Research Part C: Emerging Technologies*, 1(4), 275-291. doi:[http://dx.doi.org/10.1016/0968-090X\(93\)90002-W](http://dx.doi.org/10.1016/0968-090X(93)90002-W)
- Liwicki, M., & Bunke, H. (2002). The IAM-database: An English Sentence Database for Off-line Handwriting Recognition. *Int. Journal on Document Analysis and Recognition*, 5, 39-46.
- McAndrew, A. (2004). *An Introduction to Digital Image Processing with MATLAB*. Thomson/Course Technology.
- Rajput, N. B., Rajput, S., & Badave, S. (Octubre de 2012). Handwritten Character Recognition -A Review. *International Journal of Engineering & Technology*, Vol. 1(No. 8). Recuperado el 5 de Noviembre de 2013, de [www.ijert.org](http://www.ijert.org)
- Sezgin, M., & Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*. doi:10.1117/1.1631315
- [spacefoundation.org](http://www.spacefoundation.org). (25 de Diciembre de 2013). Obtenido de <http://www.spacefoundation.org/programs/space-technology-hall-fame/inducted-technologies/digital-image-processing-medical>
- Toselli, A. H. (2004). Reconocimiento de Texto Manuscrito Continuo. (*Tesis Doctoral*). Valencia: Univesidad Politécnica de Valencia. Recuperado el 5 de Noviembre de 2013, de <http://www.dsic.upv.es/docs/bib-dig/tesis/etd-11262003-133334/Toselli04.pdf>
- Vidal, E., Toselli, A. H., & Romero, V. (Julio de 2008). Procesado y transcripción de texto manuscrito. *Actualidad TIC*(14), 32.